

TwinCAT OPC server: OPC General

OPC General

OPC Definition

OPC stands for "Openness Productivity and Collaboration" and describes an initiative for standardisation of data exchange in automation.

Beckhoff is a active member of the OPC Foundation. (More information about OPC may be obtained under www.opcfoundation.org)

OPC Interface

Formerly, applications such as operating and monitoring software were forced to use the differing access procedures of each controller manufacturer. OPC is the specification of a universal software interface that is totally independent of the manufacturer. This opens up the possibility of gaining access in a unified manner to process data from any manufacturer.

OPC Server

An OPC server is a software component that offers a specified OPC interface to other applications.

TwinCAT OPC server: Features

Overview

- OPC Specification 1.0a and 2.0
- Free choice of TwinCAT devices (ADS devices): PLC / IO / NC / CAM / ...
- Access via TwinCAT PLC / TwinCAT I/O or bus controller to all the fieldbusses supported by TwinCAT (Lightbus / Profibus DP / Interbus-S / CANopen / DeviceNet / ...)
- Access to TwinCAT variables by address / by name / by generic ADS IndexGroup and IndexOffset
- Optimal access to data from the OPC server on TwinCAT variables
- Simulation mode (sinus, ramp, random) for variables
(i.e. testing an OPC client/server communication without access to physical I/O)
- Easy, one-time only configuration by XML

Features of the evaluation version

The evaluation version is a fully functional version with a 30 days time limitation. The OPC server then switches to its internal simulation mode. All the process values are then simulated, and there is no longer any access to the ADS devices.

The time limit in the evaluation version can be removed by purchasing a [TwinCAT OPC software licence](#), and by the entry of a valid registration key.

Integration into the TwinCAT system

The OPC server creates the connection to the TwinCAT system as a gateway via ADS, and offers OPC clients the specified OPC-DA / AE interface.

8/7/2006

Registration instructions

During the installation

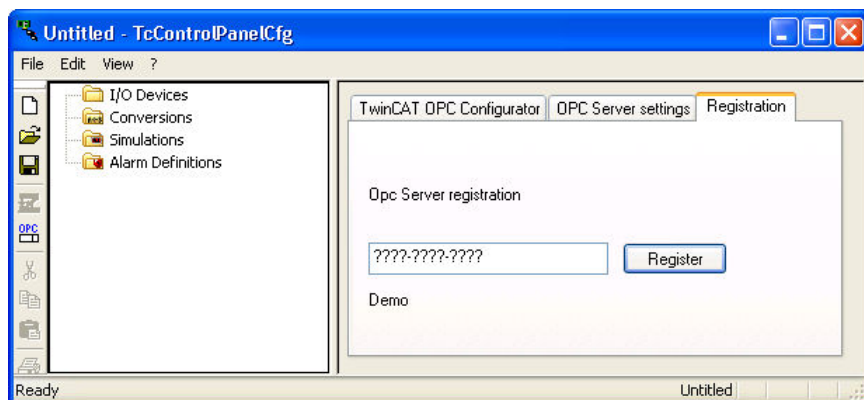
When installing TwinCAT OPC, the registration key (e.g. K123-4567-8901) can be entered into the appropriate field in the setup program. This removes the time restrictions placed on the evaluation version.

If no key is available at the time of installation, please enter "DEMO" as the key. The time-limited (30 days) evaluation version of the TwinCAT OPC will be installed.

Subsequent registration of an evaluation version

Start the program TwinCAT OPC Configurator TcOpcCfg.exe and select tab "**Registration**".

Please enter a valid registration number into the appropriate field and pree "**Register**". The time limit is removed from the evaluation version.



TwinCAT OPC Server: OPC Compliance certification

OPC Compliance Certification

The OPC foundation offers members of the organisation an compliance test tool. This tool tests the functionalities and interfaces of the OPC server. For OPC clients is no test tool available.

Testing a server should be done by competent personal only, besides the results could be forged.

Example:

In one test case, the OPC compliance tool writes variables in the OPC Server, reads them afterwards back and compares the values. If these test variables would be PLC variables, which changes cyclically in the PLC, these test case would be failed.

Compliance certification

These and further current OPC certificates are listed on the web site of the OPC foundation:

<http://www.opcfoundation.org>

OPC Data Access 2.05a Compliance Test Result

Server:	Beckhoff.TwinCATOPcServerDA (Local Server)
Compliance Test Result:	Compliant
Date:	11/6/2003 11:59:31 PM
Company:	Beckhoff Industrie Electronic
OS version:	Windows NT 5.1.2600 Service Pack 1
Version info:	<i>Compliance Test Tool:</i> V 2.0.2.1105 <i>OPC Server Major Version:</i> 4 <i>OPC Server Minor Version:</i> 1 <i>OPC Server Build Number:</i> 27 <i>OPC Server Vendor Info:</i> Beckhoff TwinCAT OPC Server4
Optional interfaces (supported):	IOPCBrowseServerAdressSpace IPersistFile
Optional interfaces (unsupported):	IOPCServerPublicGroups IOPCPublicGroupStateMgt
Data types for logical test (available):	VT_BOOL VT_I2 VT_I4 VT_R4 VT_R8 VT_BSTR VT_UI1 VT_I1 VT_UI2 VT_UI4 SAFEARRAY OF VT_I2 SAFEARRAY OF VT_I4 SAFEARRAY OF VT_R4 SAFEARRAY OF VT_R8 SAFEARRAY OF VT_UI1 SAFEARRAY OF VT_UI2 SAFEARRAY OF VT_UI4
Data types for logical test (unavailable):	VT_DATE VT_CY SAFEARRAY OF VT_BOOL SAFEARRAY OF VT_DATE SAFEARRAY OF VT_BSTR SAFEARRAY OF VT_I1 SAFEARRAY OF VT_CY
Number of supported groups:	>= 10
Not available for test:	
Number of test items:	2 Readable items 2 Writeable items 54 Read/Writeable items

TwinCAT OPC Server

TwinCAT OPC Server : Optimisations

In case of trouble with OPC-Server performance please check following steps :

1. Minimise OPC-Namespace in TwinCAT-OPC-Server :

- OPC-Server should provide required variables in his namespace which are requested by an opc-client.

It makes no sense to provide 700.000 PLC variables in the OPC-namespace but the opc-client just exchange values from 10.000 variables.

The size of OPC-namespace will cause a longer opc-server-startup-time and will cause a high memory consumption .

Option 1 :

Export just part of PLC variables required for OPC :

In the TwinCAT PLC programming environment, select "Project"->"Option"->"Symbolconfiguration". Activate "Create Symbol", and use the mouse to select under "Configure Symbolfile..." the areas that you want to export (e.g. POU's and global variables).

After compiling the PLC-project the new symbolfile "<yourProjectName>.sym" will be in the same folder as your plc-project.

REMARK:

- Each compiling of the PLC-project will automatically update the symbolfile.
- The OPC-Server imports structure and array-elements out of *.SYM files.

Option 2 :

Mark PLC-variables in the PLC-code to be relevant for OPC

This way is based on the TwinCAT export file *.tpy.

[Detailed info](#)

2. OPC-Server : Running as EXE or DLL ?

- Arguments for working with EXE :
 - One single OPC-client will access data from TwinCAT-OPC-Server
 - The OPC-Client is installed on a different PC , the network has to be crossed via DCOM.
- Arguments for working with DLL :
 - More than one OPC-Client will communicate with the OPC-Server : Instead one single OPC-Server shares his CPU time to multiple opc-clients, the OPC-Server should work as a DLL : As a result each OPC-Client will have "his personal" instance of the OPC-Server with each opc-server having a separate ADS-communication channel into the ADS-device like the PLC.

3. OPC-Server via DCOM

- The installation of BECKHOFF OPC-components consist of two parts : The native OPC-Server from Beckhoff and so called "OPC common components" which comes from the OPC-Foundation.

The installation of BECKHOFF-OPC-products have been redesigned to integrate the official OPC-components and install them in the recommended way.

Change of installation : 07/2004, new installation for TwinCAT OPC-Server V4.1.0.37

TwinCAT OPC server: Configuration

Overview

An OPC server represents a standardised interface for the management of process data. The process data available in the TwinCAT system must therefore be known to the OPC server, or must be made known to it at the time of configuration.

To represent this "hierarchical process space" clearly, the "Devices" are subdivided into subsidiary items.

The OPC client can browse through this representation and use it for the server's actual configuration. The TwinCAT OPC server supports the optional browser OPC interface.

Note:

This hierarchical display is not to be confused with the configuration of the OPC server that exists at run-time. The run-time configuration of the OPC server, i.e. the creation of groups, specification of the refresh time, the insertion of tags etc. is performed dynamically by the OPC client.

General

The advantage of configuration via XML with links to the symbols of the ADS devices is that this configuration only has to be carried out once. Even if, for instance, the PLC project changes (e.g. if variables are deleted, or if new variables are defined) it is not necessary to change the configuration in the OPC server: each time the OPC service starts, the namespace of the OPC server is correctly built up.

Device Definition

A device represents a TwinCAT ADS device (e.g. a PLC run-time system) that is uniquely identified by the <AdsAmsServerNetID> and the <AdsAmsPortID>.

Further information about the concept of devices in TwinCAT can be found in the general TwinCAT ADS documentation.

Configuration of the TwinCAT OPC server

- [Receiving data from the TwinCAT PLC](#): Configuration by variable import from TwinCAT PLC control
- [Receiving Data from the TwinCAT I/O task](#): Configuration by variable upload from the TwinCAT I/O task

Configuration of the TwinCAT OPC Server CE

In general, the configuration of the TwinCAT OPC Server CE is similar to the configuration of the TwinCAT OPC Server. The most important information is given here:

- Set the active OPC XML configuration:
 - PC: The TcOpcStartup.exe tool saves the current OPC XML configuration file in the registry
 - CE: By default, the file "\\hard disk\\TwinCAT\\OPC\\CurrentConfigOpc.xml" is entered in the registry as the current OPC XML configuration file
- The OPC XML file for CE works with the same XML scheme
- The OPC XML file must allow for the fact that paths are written using a different notation under CE:
 - PC example: <AutoCfgFile>C:\\Project123\\Test.tpy</AutoCfgFile>
 - Example CE: <AutoCfgFile>\\hard disk\\project\\Test.tpy</AutoCfgFile>

TwinCAT OPC server: Properties

The following information applies to the OPC server running on a PC platform. This information is not relevant when running on a CE platform

OPC Server as EXE / DLL (out-of-process / in-process)

TwinCAT OPC V4.1 is available in the form of an "out-of-process" server (with EXE as the file extension) and as an "in-process" server (with DLL as the file extension).

ProgID

The OPC client requires the ProgID to startup communication with TwinCAT-OPC-Server.
For TwinCAT-OPC-Server V4.1 this ProgID is "BECKHOFF.TwinCATOpcServerDA".

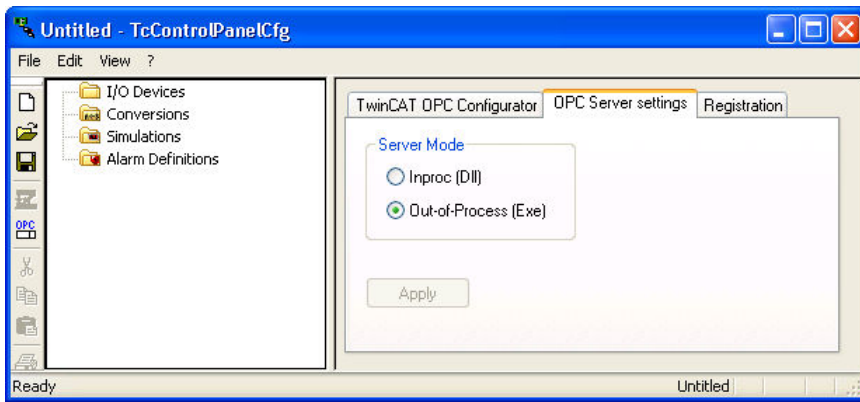
The ProgID is identical for both types, the EXE or the DLL version.

Diagnose : Which type (EXE or DLL) is active ?

Start the TwinCAT-OPC-Configurator "Start - All Programs - TwinCAT System - TwinCAT OPC - TwinCAT OPC Configurator"



Select the tab "OPC Server settings". See current active type of "Server Mode" .



Activating the TwinCAT OPC server as an in-process server (DLL)

Select "Server Mode" "**Inproc (DLL)**" and press "**Apply**".

Result :

The TwinCAT OPC server now runs as a DLL in the process space of the respective OPC client. The TwinCAT OPC server is therefore not visible as an independent process in the task manager.

The main advantage of this version is seen when a number of OPC clients are working at the same time with TwinCAT OPC server. In that case, each OPC client is given its own OPC server for its "personal use". Each OPC client then has its own independently operating OPC channel to the TwinCAT devices. Most advantage will be noticed for write-requests via OPC-Server into ADS-device.

Activating the TwinCAT OPC server as an out-of-process server (EXE)

Select "Server Mode" "**Out-of-Process (EXE)**" and press "**Apply**".

Result :

The TwinCAT OPC server now operates as an EXE program, and each OPC client works with the single, shared instance of the OPC server. In this version the TwinCAT OPC server is visible as an independent process in the task manager.

If a number of OPC clients are working at the same time with the TwinCAT OPC server, all the queries are handled by a single instance of the OPC server. This single instance eliminates or minimize double requests of variables for one single ADS device.

DCOM:

1. We do not advise the use of DCOM connections.
2. If, however, a DCOM connection is essential, it is implemented with the type "Out-of-process" OPC server.

TwinCAT OPC server: Configuration

Receiving data from the TwinCAT PLC

1st Step TwinCAT PLC

TwinCAT 2.7 (or higher):

In the TwinCAT-PLC development environment, select "Project" ->"Options" ->"Symbol configuration". Activate "Create symbol entry" there, and under "Configure symbol file.." select the areas that you want to export using the mouse. (e.g. POU's and global variables).

You will shorten the OPC server's loading time (creation of the namespace) if, as far as possible, you only mark those areas that you will in fact need in the OPC server.

After the PLC project has been compiled, there will now be a symbol file in the PLC project directory with the name structured as "<project name>.sym".

NOTE:

- The *.SYM file is automatically replaced each time the PLC project is compiled.
- The OPC server also imports structure and array elements from SYM files.

TwinCAT 2.8 (or higher):

After the PLC project has been compiled, there is always a file in the PLC project directory with the name structured as "<project name>.tpy".

NOTE:

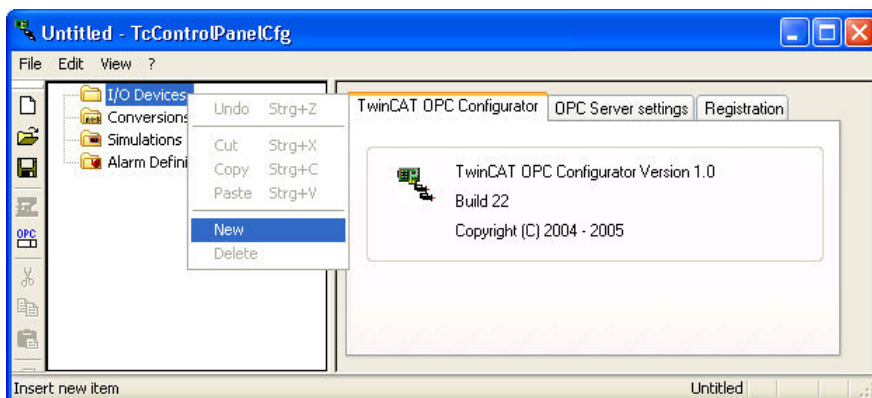
- The *.tpy file is automatically replaced each time the PLC project is compiled.
- With adding simple information done by the declaration of the PLC-variable you can mark a PLC variable to be known by the OPC-Server. This helps to reduce the OPC-namespace.
- Today the OPC server only imports global variables and does not import structures. Requiring structures and non-global variables please work on symbol-file <project name>.sym".

2nd step: Configuration of TwinCAT OPC Server

Start the TwinCAT-OPC-Configurator "Start - All Programs - TwinCAT System - TwinCAT OPC - TwinCAT OPC Configurator"

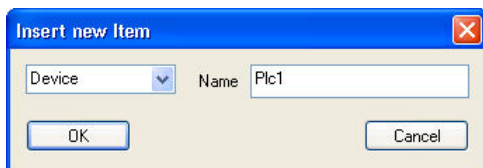


Select "I/O Devices" with right click and then menu "New".

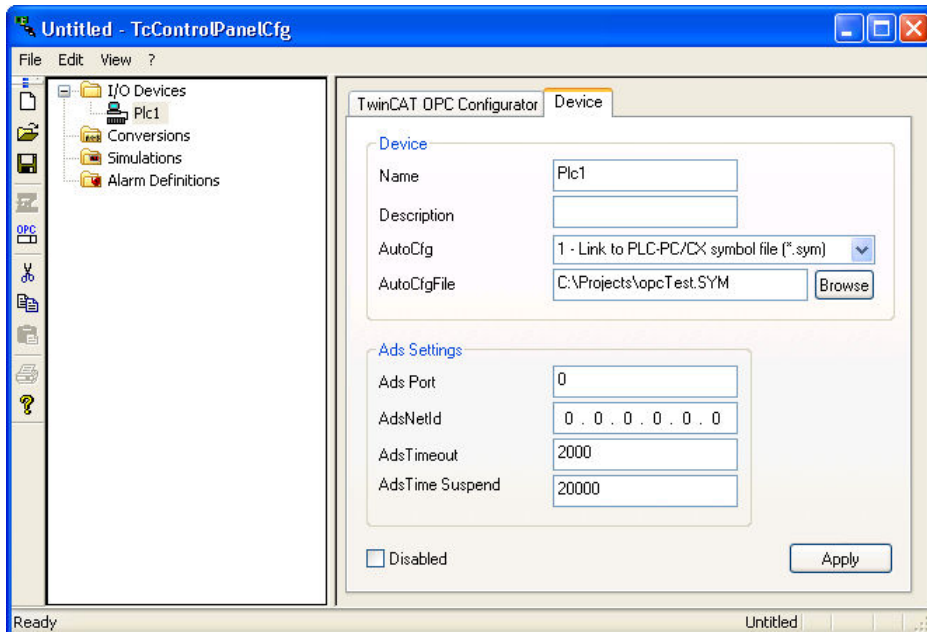


For "Name" please enter a name for this alarm definition.
This name has to be OPC conform, no special characters allowed.

Sample: The TwinCAT-PLC runtime-system 1 has a PLC variable "temperatur".
Definition of devicename with "Plc1" will publish the PLC variable via OPC as "Plc1tTemperatur" later for OPC-Clients.



After applying with "OK" you see the next dialog with settings of the new device "Plc1"



Following settings have to be checked :

Type	Req./Opt.	Description
Name	Req	Name for device like "PLC1". OPC-Client browse this name
Description	Optional	1 : Include sym-file of TwinCAT- PLC defined under <AutoCfgSymFile> 2 : Include sym-file of TwinCAT- BCxxxx-project defined under <AutoCfgSymFile>
AutoCfg	Optional	7 : Include tpy-file of TwinCAT28-project defined under <AutoCfgSymFile> 8 : Include tpy-file of TwinCAT28-project defined under <AutoCfgSymFile>, but import only symbols with symbol-property "OPC=1"
AutoCfgSymFile	Optional	full path and name of symbol-file to be included like " C:\Test\demo.sym " or " \\User1\Test\demo.sym " or "C:\User1\Test\demo.tpy"
AdsPort	Req	Number of ADS-PortID, like 801 for first PLC-runtime system 811 for first PLC-runtime system 821 for first PLC-runtime system 831 for first PLC-runtime system 800 for PLC-runtime system on BC specific ADS-NetId, like "174.12.15.45.1.1"

AdsNetId	Optional	Note: If not specified or like "0.0.0.0.0", the OPC-Server will always communicate to AdsNetId of local ADS-router
AdsTimeout	Optional	ADS timeout in ms for this ADS device. If ADS device will not replay within this time, the OPC-Server will set the requested variables to "BAD QUALITY"
AdsTimeSuspend	Optional	ADS suspend time in ms for this ADS device, if the ads communication fails. If an ADS device is running, it does not make sense to communicate with high frequency to that device. In case of ads-communication trouble the OPC-Server will wait till next ads-request to that device.
Disable	Optional	Disables this device. OPC-Server will ignore this device.

See detailed description in appendix.

3 rd step : Check and save configuration

Save the configuration under current name or select menu "File - Save As..." to insert a new name.
The OPC-Configurator will save the XML based file.

After saving the OPC-configuration file we get the option to activate this configuration to be the configuration for next startup of OPC-Server.



After selecting "Yes" this current configuration will be the start-configuration of the opc-server. With next start of OPC-Server this new configuration will be active.



NOTE:

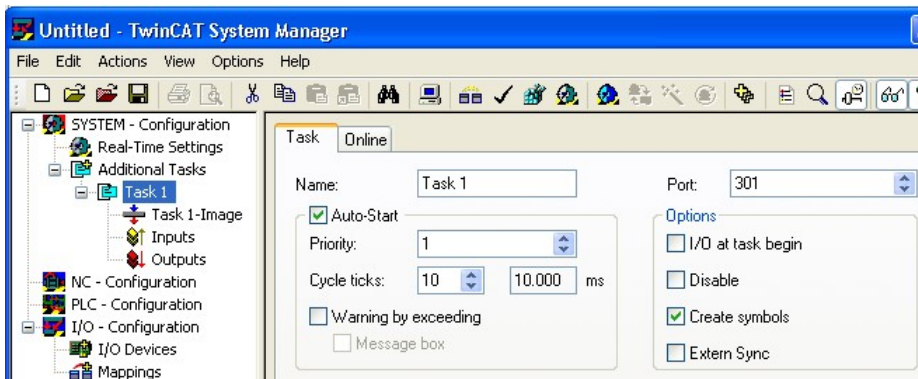
If you activated a new OPC-configuration, but the OPC-Server still has old namespace and settings the reason is that the OPC-Server was not stopped and restarted.

TwinCAT OPC server: Configuration

Receiving Data from the TwinCAT I/O task

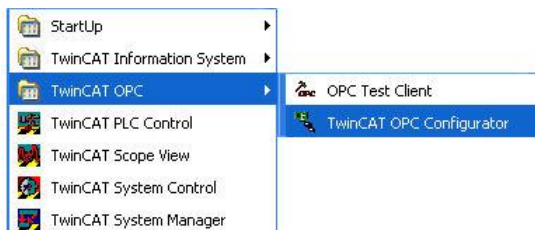
1st Step TwinCAT I/O task configuration

In the TwinCAT System Manager navigate through "System configuration ->Additional tasks -> Task x". Once there, activate "Create symbol entry".

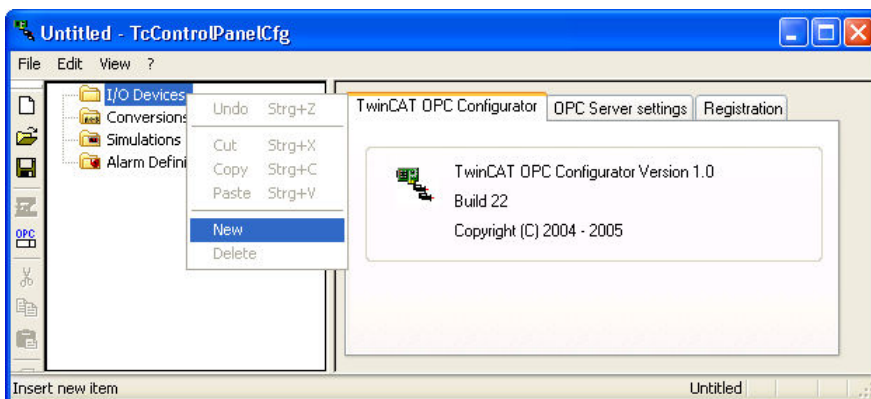


2nd Step Import into the TwinCAT OPC server

Start TwinCAT-OPC-Configurator "Start - All Programs - TwinCAT System - TwinCAT OPC - TwinCAT OPC Configurator"



In the left navigationtree please select "I/O Devices" with a right click the menu item "New".



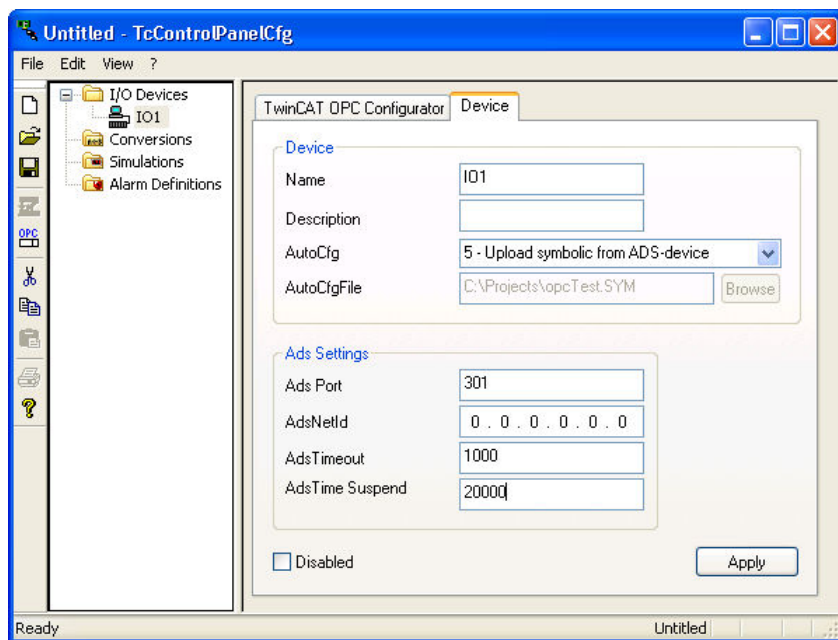
Enter a name for that ADS device in field "Name".

The name has to be OPC conform. Special characters are not allowed.

The device-name will be part of the ItemID later.

Sample: TwinCAT-PLC was defined with a variable "Temp". Choosing the devicename as "IO1" the opc-server present the variable as "IO1.Temp".

With click on "OK" we get the new dialog for further settings of device :



Following settings have to be checked :

Type	Req./Opt.	Description
Name	Req	Name for device like "IO1". OPC-Client browse this name
Description	Optional	
AutoCfg	Optional	5 : Upload symbolic from ADS-device
AutoCfgFile	Optional	will be ignored
AdsPort	Req	Number of ADS-PortID, like 301 for first IO task 302 for next IO task ... specific ADS-NetId, like "174.12.15.45.1.1"
AdsNetId	Optional	Note: If not specified or like "0.0.0.0.0", the OPC-Server will always communicate to AdsNetId of local ADS-router
AdsTimeout	Optional	ADS timeout in ms for this ADS device. If ADS device will not replay within this time, the OPC-Server will set the requested variables to "BAD QUALITY"
AdsTimeSuspend	Optional	ADS suspend time in ms for this ADS device, if the ads communication fails. If an ADS device is running, it does not make sense to communicate with high frequency to that device. In case of ads-communication trouble the OPC-Server will wait till next ads-request to that device.
Disable	Optional	Disables this device. OPC-Server will ignore this device.

See detailed description in appendix.

3 rd step : Check and save configuration

Save the configuration under current name or select menu "File - Save As..." to insert a new name.
The OPC-Configurator will save the XML based file.

After saving the OPC-configurationfile we get the option to activate this configuration to be the configuration for next startup of OPC-Server.



After selecting "Yes" this current configuration will be the start-configuration of the opc-server. With next start of OPC-Server this new configuration will be active.



NOTE:

If you activated a new OPC-configuration, but the OPC-Server still has old namespace and settings the reason is that the OPC-Server was not stopped and restarted.

TwinCAT OPC Server: Configuration

Configuration OPC Alarm & Event

TwinCAT OPC-Server offers OPC-specification OPC AE.

Sample

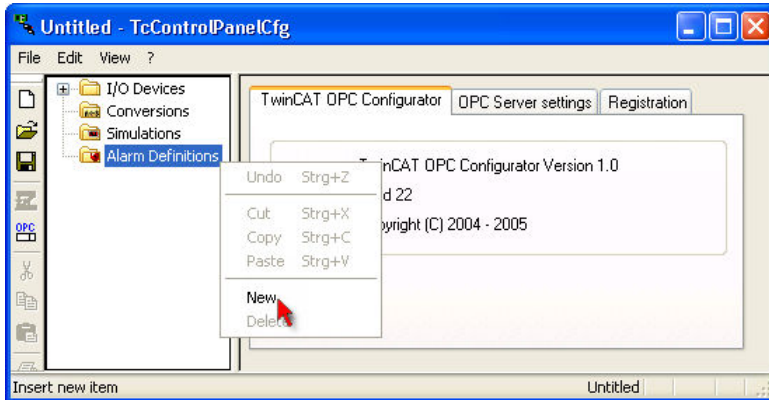
See this [complete sample](#) including PLC-project and the OPC-AE-configuration.

1. Step: Configuration of Alarms for TwinCAT OPC Server AE

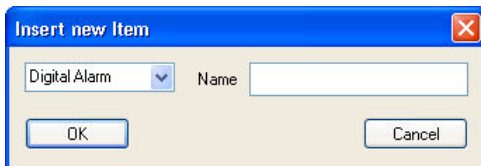
Start TwinCAT-OPC-Configurator "**Start - All Programs - TwinCAT System - TwinCAT OPC - TwinCAT OPC Configurator**"



Navigate to "**Alarm Definitions**", right click and select menu "**New**".

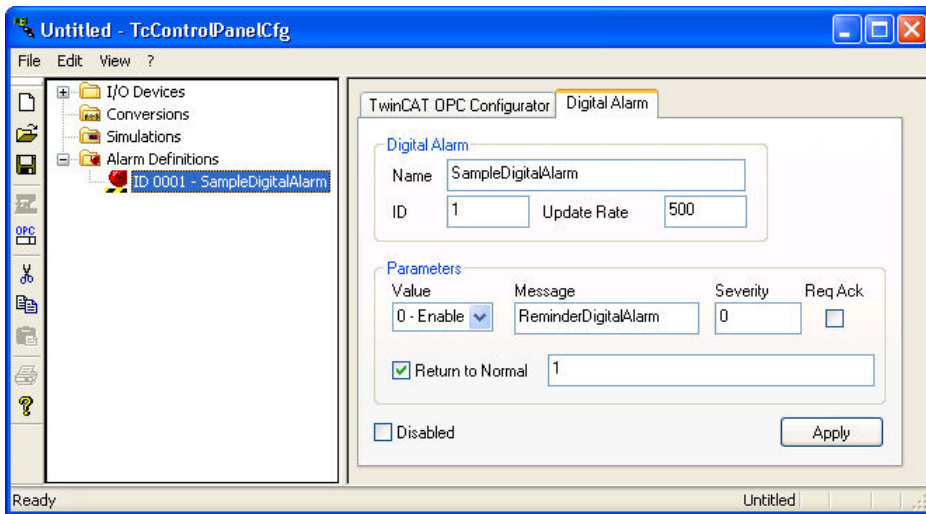


- Select "**Digital Alarm**" or "**Limit Alarm**".
- For "**Name**" please enter a name for this alarm definition.
This name has to be OPC conform, no special characters allowed.



Option 1 : Digital Alarm

After selecting "**Digital Alarm**" and entering "OK" you get into the dialog to configure in details your new "**Digital Alarm**".

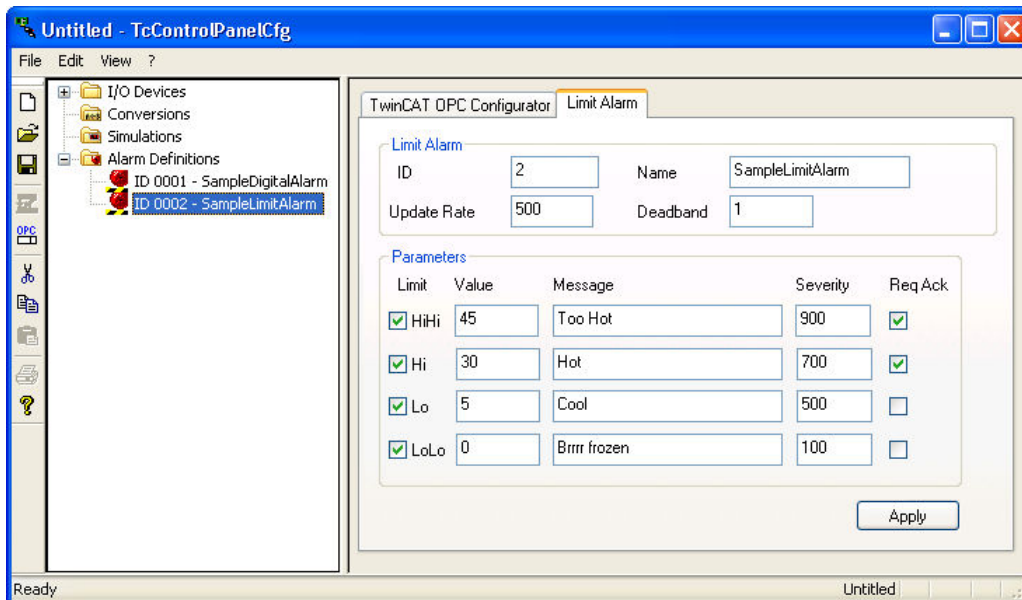


Following settings can be done :

Type	Req./Opt.	Description
Name	Required	Name for alarm
Alarm ID	Required	Unique Alarm-ID Later, this Alarm-ID has to be linked to a PLC variable in the PLC-control
Update Rate	Required	Update rate of alarm
Value	Req	
Message	Optional	Message
Severity	Optional	
Req. Acknowledge	Optional	
Return to Normal		
Disable	Optional	Disables this alarm. OPC-Server will ignore this alarm.

Option 2 : Limit Alarm

After selecting "**Limit Alarm**" and entering "OK" you get into the dialog to configure in details your new "**Limit Alarm**".



Following settings can be done :

Type	Req./Opt.	Description
Name	Required	Name for alarm
Alarm ID	Required	Unique Alarm-ID Later, this Alarm-ID has to be linked to a PLC variable in the PLC-control
Update Rate	Required	Update rate of alarm
Deadband	Required	Update rate of alarm
HiHi	Req	
Value	Optional	Message
Message	Optional	
Severity	Optional	
Req. Acknowledge		
Disable	Optional	Disables this alarm. OPC-Server will ignore this alarm.

2. Step: Check and save configuration

Save new configuration by selecting menu "File - Save As..."

After successfully creating and saving of new OPC-configurationfile the next dialog offers to activate this new configuration.



Select "OK" to activate the configuration. With next startup the OPC-Server will work with new settings.



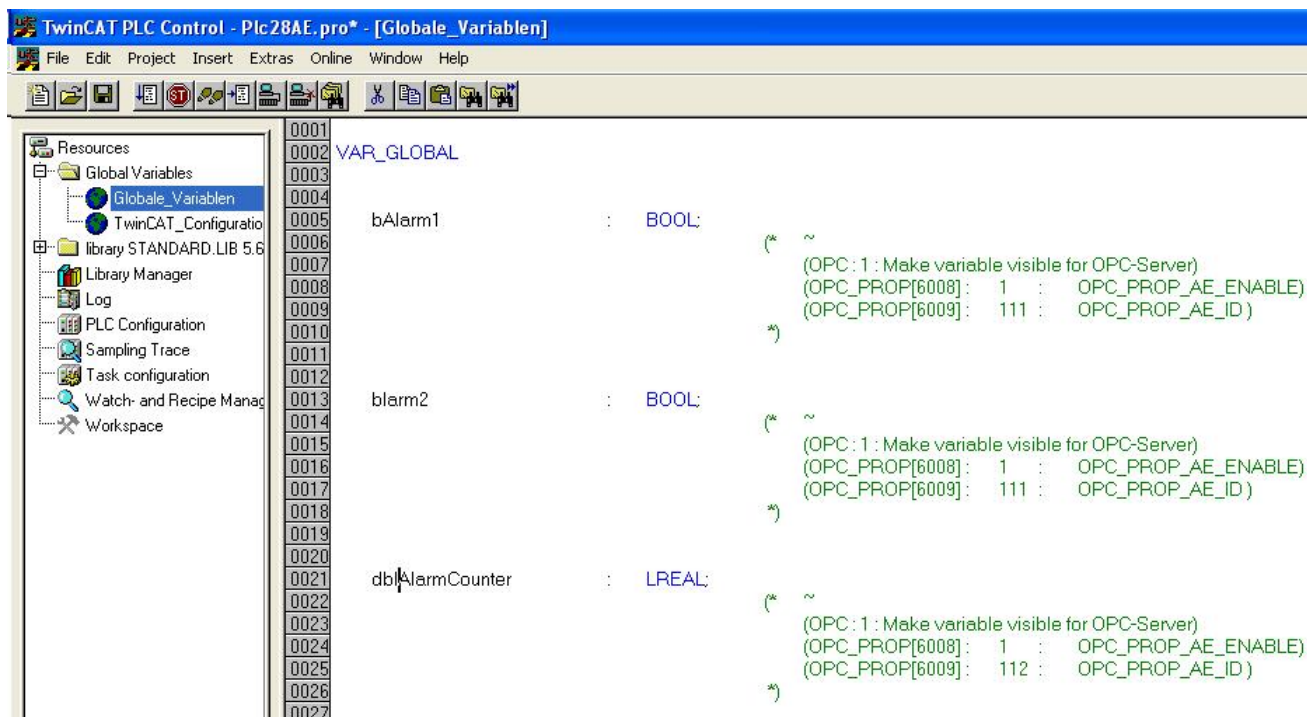
NOTE:

In case of opc server offers old previous configuration to opc-clients but a new configuration was activated this means, that OPC-server was not restarted to load fresh configuration.

3. Step: Link of PLC-Variable with Alarm-Type

With previous steps we defined a "Alarm-Template" with detailed information about alarm handling. Now we define, which PLC variable should be handled like an alarm.

This link is configured in PLC-Control as and additional information for IEC1131-variable declaration.



Take care on following settings :

Add a comment behind PLC-variable declaration. This comment will be interpreted by the opc-server and links this plc-variable to a previous defined alarm-template.

The alarm-template with the unique Alarm-ID was configured in step 1.

NOTE:

Today just global PLC-variables can be linked to Alarm-Templates. Elements of structures or arrays can not be configured as an OPC-alarm. Future versions of OPC-Server will not have this limitation.

Sample :

- PLC-Variable of Typ BOOL **bAlarm1** and **bAlarm2** are defined, both variables are linked to alarm-template with Alarm-ID 111. The OPC-AE-Server will publish this variable as an AE-variable.
- PLC-Variable of Typ LREAL **dblAlarmCounter** is a OPC-AE variable as Limit-Alarm with alarm-template 112.

Note, the syntax is important :

```
bAlarm1 : BOOL; (* ~
              ( OPC : 1 : Enables this variable to be an OPC variable)
              ( OPC_PROP[6008] : 1 : Enables the Alarm)
              ( OPC_PROP[6009] : 111 : Description of Alarm)
              *)
```

Each time compiling the PLC-project the PLC-Control will create the file <PLC-projectname>.tpy. This XML based file contains information about PLC-variables and their link to OPC-AE.

The OPC server will analyze this information of the TPY file., so configure the OPC-Server to know the file <PLC-projectname>.tpy [Details](#)

TwinCAT OPC Server: Configuration

Configuration OPC conversions

The OPC server offers the feature to online convert process data.

The OPC-Server takes care of online-conversion in both communication directions :

- Processvalue --> Communication to OPC-Server --> conversion within OPC-Server --> Communication to OPC-Client --> OPC-Client
- OPC-Client --> Communication to OPC-Server --> conversion within OPC-Server --> Communication to process --> Processvalue

Sample :

- KL3202 offers process value of temperatures in unit 1/10 Grad Celsius, e.g. 200.
- The opc-server is configured with a conversion "Factor 10" which is linked to this plc-variable.
- The opc-client gets the process value 20 Grad.
- If opc-client writes a new value 25 Grad celsius to OPC-server, the server will convert this to 250 and writes this value to the PLC.

Sample

The [complete sample](#) contains a PLC project and an OPC conversion definition.

NOTE:

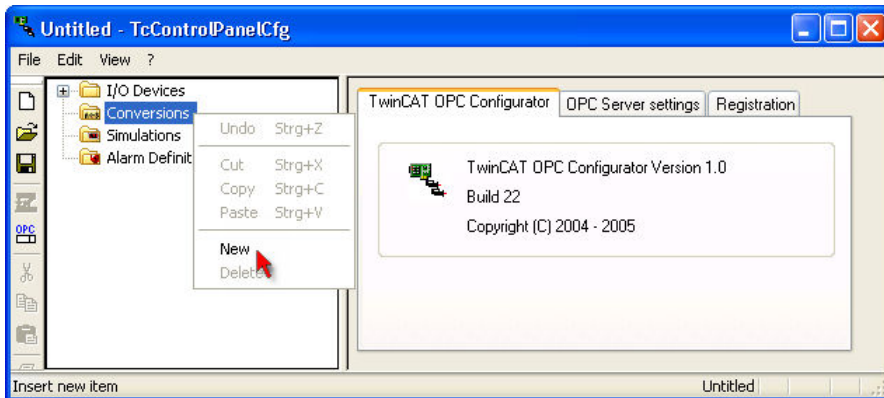
OPC-Server presents a converted variable as Datatyp "VT_R8 / Double" to OPC-client - independend of numerical datatyp in the PLC.

1. Step: Configuration of conversion for TwinCAT OPC Server

Start TwinCAT-OPC-Configurator "Start - All Programs - TwinCAT System - TwinCAT OPC - TwinCAT OPC Configurator"

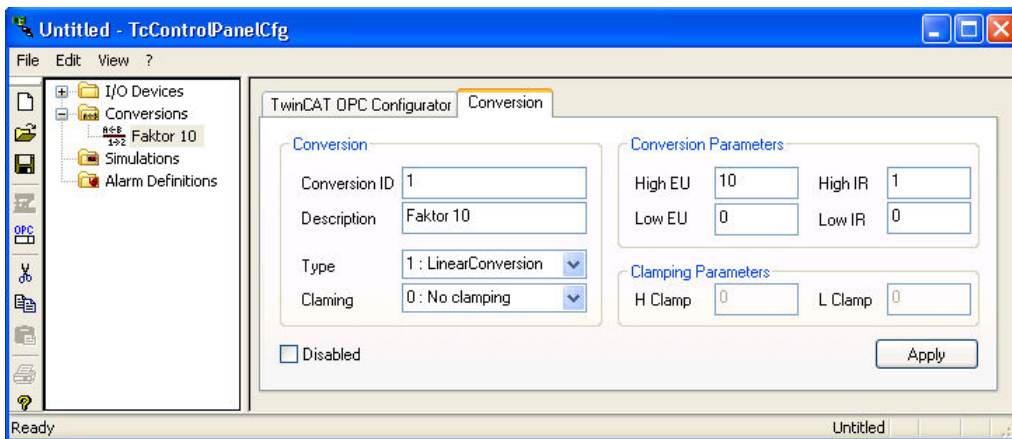


Navigate to "**Conversions**", right click and select menu "**New**".



For "**Name**" please enter a name for this alarm definition.
This name has to be OPC conform, no special characters allowed.

With "OK" you see the dialog to configure in detail your new "**Conversion**".



Following settings can be done :

Type	Req./Opt.	Description
Conversion ID	Required	
Type	Required	
Clamping		
High EU	Req	
Low EU	Optional	
High IR	Optional	
Low IR	Optional	
H		
Clamping		
L		
Clamping		

Disable	Optional	Disables this conversion. OPC-Server will ignore this conversion.
---------	----------	---

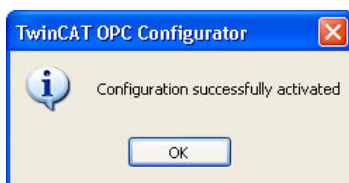
Step: Check and save configuration

Save new configuration by selecting menu "File - Save As..."

After successfully creating and saving of new OPC-configurationfile the next dialog offers to activate this new configuration.



Select "OK" to activate the configuration. With next startup the OPC-Server will work with new settings.



NOTE:

In case of opc server offers old previous configuration to opc-clients but a new configuration was activated this means, that OPC-server was not restarted to load fresh configuration.

3. Step: Link of PLC-Variable with Conversion-template

With previous steps we defined a "conversion-Template" with detailed information about conversion handling.

Now we define, which PLC variable should be handled with conversion.

This link is configured in PLC-Control as and additional information for IEC1131-variable declaration.

Take care on following settings :

Add a comment behind PLC-variable declaration. This comment will be interpreted by the opc-server and links this plc-variable to a previous defined conversion-template.

The conversion-template with the unique conversion-ID was configured in step 1.

NOTE:

Today just global PLC-variables can be linked to conversion-templates. Elements of structures or arrays can not be configured as an OPC-conversion.

Future versions of OPC-Server will not have this limitation.

Sample :

- PLC-Variable of Typ INT **intMemCounterConversion** is an OPC-DA variable which is converted by the opc-server as defined in conversion template 102.

Note, the syntax is important :

```
intMemCounterConversion : INT;  (* ~
                                (OPC : 1 : available for OPC-clients)
                                (OPC_PROP[6010] : 1 : OPC_PROP_CONV_ENABLE)
                                (OPC_PROP[6011] : 102 : OPC_PROP_CONV_ID)
                                *)
```

Each time compiling the PLC-project the PLC-Control will create the file <PLC-projectname>.tpy. This XML based file contains information about PLC-variables and their link to the OPC-server.

The OPC server will analyze this information of the TPY file., so configure the OPC-Server to know the file <PLC-projectname>.tpy [Details](#)

TwinCAT OPC Server: Configuration

Configuration OPC Simulation

The OPC-Server offers to simulate process-values. In this case no communication to the ADS-device (like PLC) is done instead simulated values are send to OPC-client.

Different simulation templates like sinus, ramp, random etc. are available to be configured with detailed behaviour (amplitude, start /stop values).

The feature "Simulation" ist available in OPC Server version V4.0.x.x.

OPC-Server version V4.1.x.x offers a multiple more performant access to TwinCAT-devices, the simulation-mode is under construction andnot supported today.

The OPC-Server offers

- simulation of all process values
- simulation of some specific process values

It is not possible to combine both simulated and real-process values.

Sample

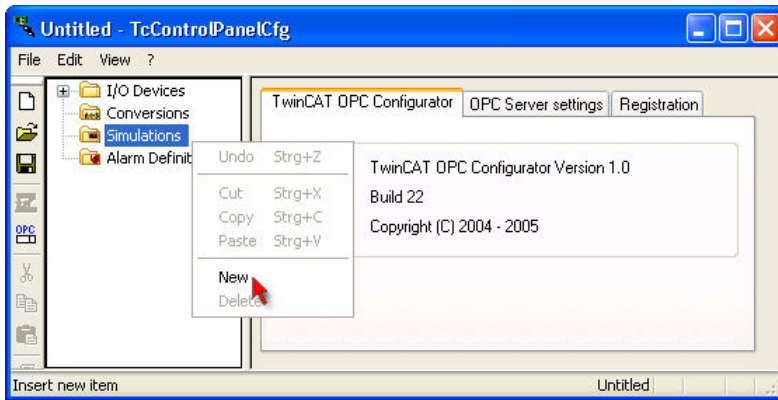
The [complete sample](#) contains both, the PLC project and the definition of OPC-simulation.

1. Step: Configuration of simulation-templates for TwinCAT OPC Server

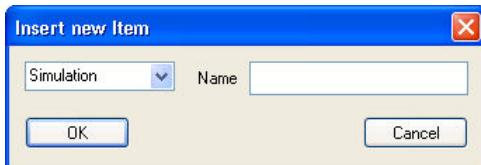
Start TwinCAT-OPC-Configurator "Start - All Programs - TwinCAT System - TwinCAT - TwinCAT OPC Configurator"



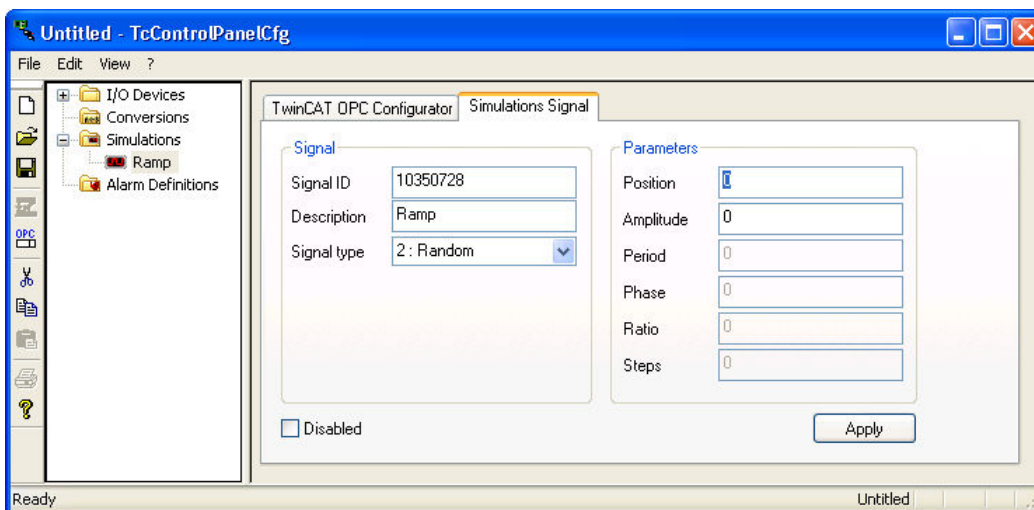
Navigate to "**Simulations**", right click and select menu "**New**".



For "**Name**" please enter a name for this simulation template.
This name has to be OPC conform, no special characters allowed.



With "OK" you get into the dialog to configure in details your new "**Simulations Signal**".



Following settings can be done :

Signal Req./Opt. Description
Type

0 :
ReadCount Optional
1 :
WriteCount Optional
2 :
Random Optional

3 : Ramp Optional
 4 : Sine Optional
 5 : Square Optional
 6 : Triangle Optional
 7 : Stepp Optional

Disable Optional Disables this signal. OPC-Server will ignore this signal.

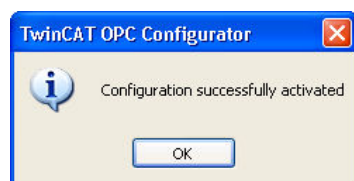
2. Step: Check and save configuration

Save new configuration by selecting menu "File - Save As..."

After successfully creating and saving of new OPC-configurationfile the next dialog offers to activate this new configuration.



Select "OK" to activate the configuration. With next startup the OPC-Server will work with new settings.



NOTE:

In case of opc server offers old previous configuration to opc-clients but a new configuration was activated this means, that OPC-server was not restarted to load fresh configuration.

3. Step: Link of PLC-Variable with Alarm-Type

With previous steps we defined a "simulation-template" with detailed information about simulation behaviour. Now we define, which PLC variable should be handled like an simulation.

This link is configured in PLC-Control as and additional information for IEC1131-variable declaration.

Take care on following settings :

Add a comment behind PLC-variable declaration. This comment will be interpreted by the opc-server and links this plc-variable to a previous defined simulation-template.

The simulation-template with the unique Simulation-ID was configured in step 1.

NOTE:

Today just global PLC-variables can be linked to Simulation-Templates. Elements of structures or arrays can not be configured as an simulated variable.

Future versions of OPC-Server will not have this limitation.

Sample :

- PLC-Variable of Typ LREAL **dblMemAlarmCounter** is a OPC variable linked to simulation template 6.
The OPC-Server will not access this PLC variable, but instead give simulated values back to OPC-client.

Note, the syntax is important :

```
dblMemAlarmCounter  : LREAL; (* ~
                      (OPC      : 1 : available for OPC-clients)
                      (OPC_PROP[6006] : 1 : OPC_PROP_SIMU_ENABLE)
                      (OPC_PROP[6007] : 6 : OPC_PROP_SIMU_ID)
                      *)
```

Each time compiling the PLC-project the PLC-Control will create the file <PLC-projectname>.tpy. This XML based file contains information about PLC-variables and their link to simulation templates.

The OPC server will analyze this information of the TPY file., so configure the OPC-Server to know the file <PLC-projectname>.tpy [Details](#)

TwinCAT OPC Server: Configuration

Configuration OPC Property IDs

OPC-spezification allows to add additional information to single OPC-items (opc-item means : a TwinCAT variable). These optional functionality is named **"OPC-Item-Properties"** in the OPC-spec and allows an opc-client to browse and read these additional properties.

Sample :

As an option the PLC variable "temperature" can hold addition informaton like **comment** (OPC-Property-ID 101) and also **UNIT** (OPC-Property-ID 100) which can be offered by the opc-server.

Configuration of OPC Property ID in the PLC

Add a comment behind PLC-variable declaration. This comment will be interpreted by the opc-server and will define OPC-Item-Properties.

NOTE:

Today just global PLC-variables can be hold item-properties. Elements of structures or arrays can not be configured.
Future versions of OPC-Server will not have this limitation.

Sample of PLC variable-declaration:

NOTE : All item-properties are optional.

```
ITemperatur : DWORD;
              (* ~
              (OPC : 1 : Make variable visible for OPC-Server)
              (OPC_PROP[0005] : 3 : OPC_PROP_RIGHTS, here Read AND Write Access)
              (OPC_PROP[0100] : Grad F : OPC_PROP_UNIT)
              (OPC_PROP[0101] : Demovariable : OPC_PROP_DESC)
              (OPC_PROP[0205] : We are the champions : OPC_PROP_SND)
              (OPC_PROP[0206] : ..\..\info.html : OPC_PROP_HTML)
              (OPC_PROP[0207] : ..\..\service.avi : OPC_PROP_AVI)
              *)
```


Each time compiling the PLC-project the PLC-Control will create the file <PLC-projectname>.tpy. This XML based file contains information about PLC-variables and their link to the OPC-server.

The OPC server will analyze this information of the TPY file., so configure the OPC-Server to know the file <PLC-projectname>.tpy [Details](#)

List of supported OPC-item properties of OPC-Specification :

Property Req./Opt. Description

ID

laut OPC

Spec

OPC_PROP_RIGHTS

OPC_PROP [0005] Optional 1 : OPC-Server will publish this variable with access :
OPC_READABLE
2 : OPC-Server will publish this variable with access :
OPC_WRITEABLE
3 : OPC-Server will publish this variable with access :
OPC_READABLE AND OPC_WRITEABLE

Default : 3 : ReadWrite Zugriff

OPC_PROP [0100] Optional OPC_PROP_UNIT : Unit

OPC_PROP [0101] Optional OPC_PROP_DESC : Description

OPC_PROP [0102] Optional OPC_PROP_HIEU

OPC_PROP [0103] Optional OPC_PROP_LOEU

OPC_PROP [0106] Optional OPC_PROP_CLOSE

OPC_PROP [0107] Optional OPC_PROP_OPEN

OPC_PROP [0200] Optional OPC_PROP_DSP

OPC_PROP [0201] Optional OPC_PROP_FGC

OPC_PROP [0202] Optional OPC_PROP_BGC

OPC_PROP [0203] Optional OPC_PROP_BLINK

OPC_PROP [0204] Optional OPC_PROP_BMP

OPC_PROP [0205] Optional OPC_PROP_SND

OPC_PROP [0206] Optional OPC_PROP_HTML

OPC_PROP [0207] Optional OPC_PROP_AVI

OPC_PROP [6007] Optional BECKHOFF Defined : Simulation ID

OPC_PROP [6008] Optional BECKHOFF Defined : Alarm enabled

OPC_PROP [6009] Optional BECKHOFF Defined : Alarm ID

OPC_PROP [6010] Optional BECKHOFF Defined : Conversion enabled

OPC_PROP [6011] Optional BECKHOFF Defined : Conversion ID

Description of the TwinCAT-OPC-Server XML configuration

Schemes

Namespace

Devices

Device <see [Device-Parameters](#)>

Device <see [Device-Parameters](#)>

SimulationSignals

Simulation <see [Signal-Parameters](#)>

Simulation <see [Signal-Parameters](#)>

Simulation <see [Signal-Parameters](#)>

Simulation <see [Signal-Parameters](#)>

...

Conversions

Conversion <see [Conversion-Parameters](#)>

Conversion <see [Conversion-Parameters](#)>

Conversion <see [Conversion-Parameters](#)>

Conversion <see [Conversion-Parameters](#)>

...

Device-Parameters

Type	Req./Opt.	Description
Name	Req	Name for device like "PLC1". OPC-Client browse this name
Description	Optional	
AdsPort	Req	Number of ADS-PortID, like 801 for first PLC-runtime system specific ADS-NetId, like "174.12.15.45.1.1"
AdsNetId	Optional	Note: If not specified or like "0.0.0.0.0.0", the OPC-Server will always communicate to AdsNetId of local ADS-router
AdsTimeout	Optional	ADS timeout in ms

		for this ADS device
AdsTimeSuspend	Optional	ADS suspend time in ms for this ADS device, if the ads communication fails
		0 : do not include symbol-file defined under <AutoCfgSymFile> 1 : Include sym-file of TwinCAT- PLC defined under <AutoCfgSymFile> 2 : Include sym-file of TwinCAT- BCxxxx-project defined under <AutoCfgSymFile> 5 : Upload symbolic from ADS-device 7 : Include tpy-file of TwinCAT28- project defined under <AutoCfgSymFile>
AutoCfg	Optional	
		8 : Include tpy-file of TwinCAT28- project defined under <AutoCfgSymFile>, but import only symbols with symbol-property "OPC=1" full path and name of symbol-file to be included like "C:\Test\demo.sym" or "\User1 \Test\demo.sym" or "C:\User1 \Test\demo.tpy"
AutoCfgSymFile	Optional	

Signal-Parameters

Type	Req./Opt.	Description
SignalID	Req	unique ID- number which identifies this simulation- signal
SignalDesc	Optional	
		0 : Read Count
		1 : Write Count
		2 :
		Random
		3 : Ramp
		4 : Sine
		5 : Square
		6 :
		Triangle
		7 : Step
		8 :
		Reserver
		9 : Step

Read

Count :
incremented
by one
every time
when the
item is read

Write

Count :
incremented
by one
every time
when the
item is
written

Random:

generates
random
value
within the
Amplitude
range
starting
with
Position

Ramp,**Sine,****Square,****Triangle,****Step:**

(periodical
signals)

Their time
behavior is
influenced
by Period
and Phase
parameters.
Period
specifies
the signal
frequency,
while Phase
moves the
signal
origin on
the time
axis

Square and
Triangle
signal types
have one
more
parameter:
Ratio. Ratio
defines
Triangle
signal
steepness,
or Square
signal H/L
proportions.
NumSteps
parameter
of the Step
signal
defines a
number of
steps signal
amplitude

SignalType Req

	will be divided into.
NumSteps	Optional, depends on <SignalType>
Amplitude	Optional, depends on <SignalType>
Period	Optional, depends on <SignalType>
Phase	Optional, depends on <SignalType>
Position	Optional, depends on <SignalType>
Ratio	Optional, depends on <SignalType>

Conversion-Parameters

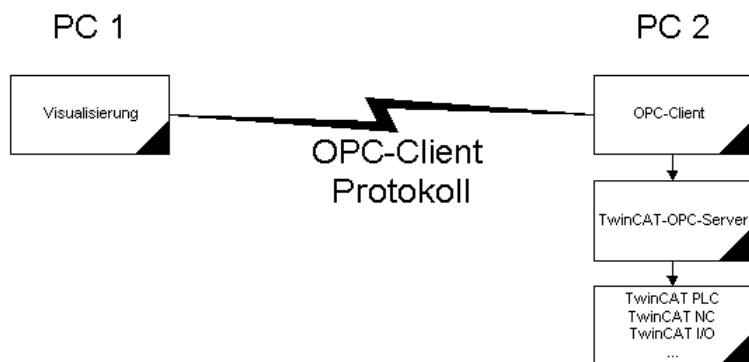
Type	Req./Opt.	Description
ConversionID	Req	unique ID-number which identifies this conversion
ConversionDesc	Optional	
ConversionType	Req	0 : NoConversion 1 : LinearConversion 0 : No clamping 1 : Clamp on EU 2 : Clamp as specified
Clamping	Optional	If clamping is active, the data value will be limited to its High clamp/EU value, when it exceeds the upper limit, and similarly with Low clamp parameter.
HighClamp	Optional, depends on <ConversionType>/<Clamping>	1.0 (Default)
LowClamp	Optional, depends on <ConversionType>/<Clamping>	0.0 (Default)
HighEU	Optional, depends on <ConversionType>/<Clamping>	engineering unit (client scale) 1.0 (Default)
LowEU	Optional, depends on <ConversionType>/<Clamping>	engineering unit (client scale) 0.0 (Default)
HighIR	Optional, depends on <ConversionType>/<Clamping>	instrument range (device scale) 10000 (Default)
LowIR	Optional, depends on <ConversionType>/<Clamping>	instrument range (device scale) 0 (Default)

TwinCAT OPC server: Data exchange over a network

Network via client protocol

Visualisation systems sometimes offer their own protocol to bridge the network. An OPC client from the visualisation system supplier is also installed on the TwinCAT controller PC for this purpose.

The data flow proceeds as follows:



Performance advantage

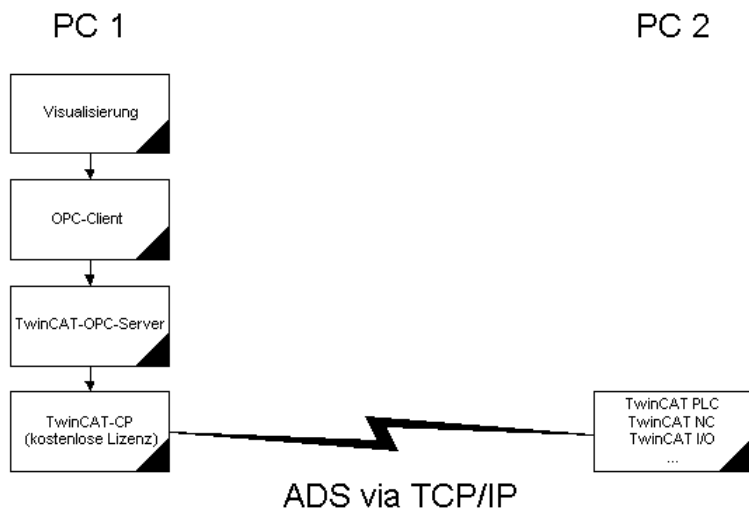
The primary advantage of this configuration is that considerably less loading is placed on the network, since only the process values that have altered are exchanged there: the OPC server checks the whole process cyclically, but only passes the changes in the process on to the OPC client.

TwinCAT OPC server: Data exchange over a network

Network via TwinCAT ADS

The TwinCAT OPC server performs the communication with the TwinCAT components (PLC run-time systems, bus terminal controllers etc.) via TwinCAT ADS. Since these services are also available in the network, the TwinCAT OPC server can also communicate with TwinCAT components that exist in the network.

You will find the [settings needed for ADS communication over a network](#) in the TwinCAT documentation.



Settings in the OPC server

An ADS communication partner is always specified by two parameters: These are what is known as the "AdsAmsNetId" (e.g. 1.2.3.4.5.6) and the "PortId" (e.g. 801 for the first PLC run-time system).

If ADS is to be used to exchange data with, for example, the first PLC run-time system on another PC, then the AdsAmsNetId of the target PC, and "801" as the PortId, must be entered in the OPC server under "Device settings".

Note

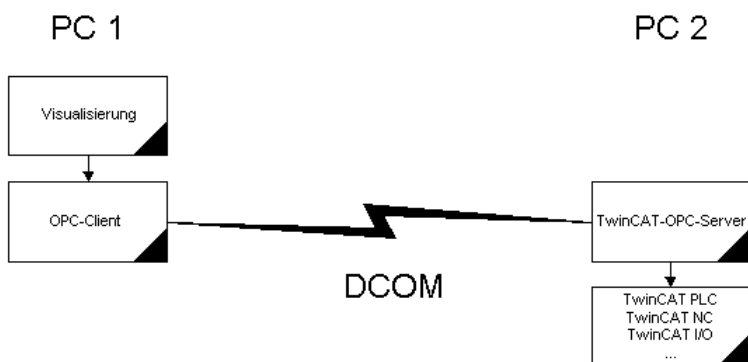
If possible, the option of using the "[Network via client](#)" protocol should be selected for performance reasons.

TwinCAT OPC server: Data exchange over a network

Network via DCOM

If you wish (or need) to allow data exchange between an OPC client and the TwinCAT OPC server over a network via DCOM, this DCOM access type must be configured beforehand.

The data flow is then as follows:



The following steps describe the necessary settings for the PC on which the TwinCAT OPC server is installed. (Use step 1 for the PC on which the OPC client is installed.)

Requirement

It is assumed that all the PCs which are to participate via DCOM in an OPC client/server connection are already set up for network operation (i.e. they have configured network cards running the TCP/IP protocol)

TwinCAT OPC server: DCOM details

General Information

This Help File was designed to give the user of component's communication through DCOM (specially OPC users) an idea on possible settings in an industrial environment. This Help File just shows possible settings of DCOM security that will make the system running. If the manufacturers or vendors of OPC products provide their own manuals, these manuals should be used instead of this Help File.

Important Notes

Before changing any settings of DCOM security, a system recovery procedure should be prepared. All settings described in this documentation should be tested in a laboratory environment, before changing machines in production.

- Save system partition including Windows Registry (Image Software)
- Only local administrators are allowed to change DCOM settings
- Test all settings in a laboratory environment before changing the production machine

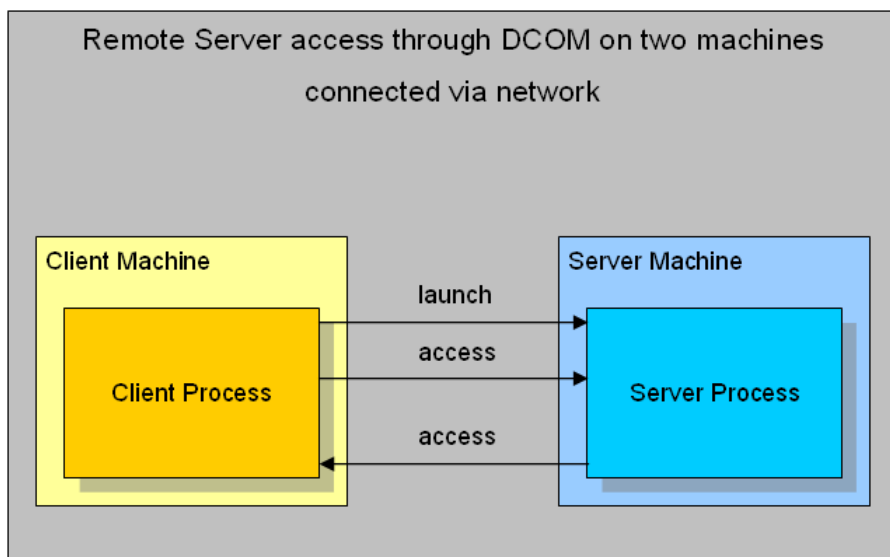
TwinCAT OPC server: DCOM details

Overview

This documentation deals with COM components that should communicate with each other. There are three different types of COM -Servers known, depending on their operational environment.

- **INPROG- Server**
The COM-Server runs in the memory area of the Client, the Server is a DLL and runs local on the same machine and in the same process.
- **LOCAL-Server**
The COM Server runs in it's own memory area, the Server is a EXE and runs local on the same machine, but in it's own process
- **REMOTE-Server**
The COM Server runs in it's own memory area, the Server is a EXE and runs remotely on a different machine and (of course) in it's own process as the Client

The communication between Client and Server process follows the rules of COM. Whenever the Client is located on one machine and the Server on an other, DCOM (Distributed COM) comes into place. The function calls between Client and Server process are checked for correct security by the operating system. The security settings for DCOM have NOTHING to do with sharing folders between two machines or network shares across a network. When it comes to DCOM-Security we talk about "inter process communication", in other words: the right to start or access a certain component.



To keep configuration simple it is assumed that two machines have the same operating system installed, are both in the same Workgroup and have the same user accounts (same person/PWD actually logged on). On the Server Machine an OPC Server is located and on the Client Machine an OPC Client was installed.

TwinCAT OPC server: DCOM details

Network Configuration

Depending on the configuration of the network different identification of Users will be performed by the operating system. Considerations regarding this issues are divided into the following topics.

- [Workgroup](#)
- [Domain](#)

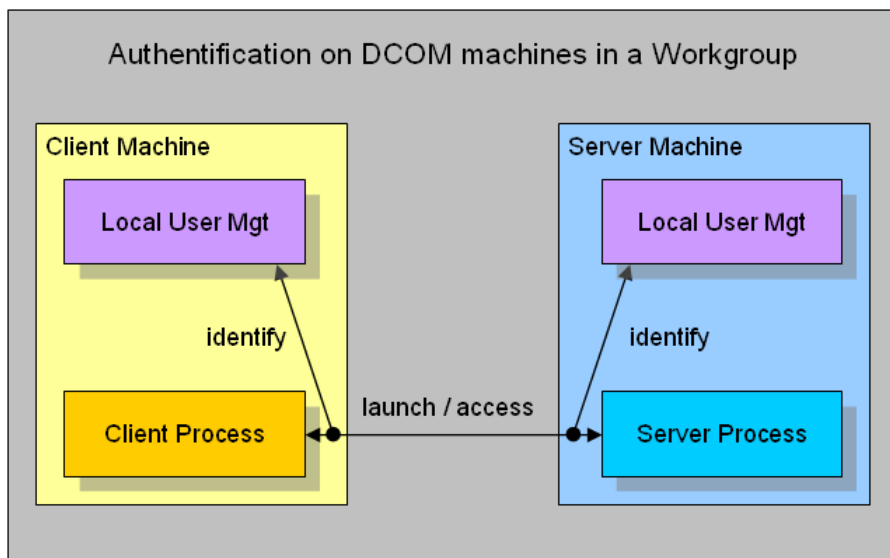
TwinCAT OPC server: DCOM details

Workgroup

For proper DCOM security settings it is essential to identify the configuration of the Client and the Server machine. If both machines are members of the same workgroup there will be no central point for user authentication. This means that every single machine must have all information on all users that should be able to remote access this node (local identification). The administrative effort will increase immensely when adding new users.

- **Workgroup**
The Client and Server Machine should be member of the same Workgroup.
- **Users Accounts**
As authentication is performed locally on every single machine, the User Accounts (Name and PWD) should be identical on the Client and the Server Machine.
- **Operation System**
The Operation System on the Client and the Server Machine should be from the same family (all NT, all 2K or all XP). When doing "mixed configuration" certain (OS specific) settings have to be taken into account.

To keep configuration simple it is assumed that two machines have the same operating system installed, are both in the same Workgroup and have the same user accounts (same person/PWD actually logged on). On the Server Machine an OPC Server is located and on the Client Machine an OPC Client was installed.



TwinCAT OPC server: DCOM details

Domain Controller

For proper DCOM security settings it is essential to identify the configuration of the Client and the Server machine. If both machines are members of the same domain there will be one central point for user authentication. If both machines are in different domains, these domains must trust each other. The administrative effort will decrease because new users will only be added to the domain.

Domain

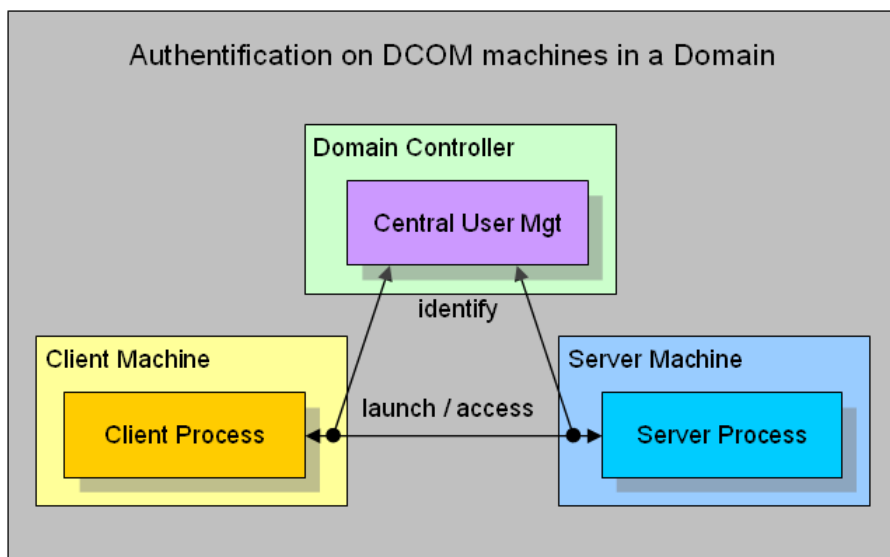
The Client and Server Machine should be member of the same Domain. Different domains must should be trust each other.

Users Accounts

Authentication is performed on the domain machine, the User Accounts (Name and PWD) or groups are used in the DCOM settings of the Client and the Server Machine.

- **Operation System**
The Operation System on the Client and the Server Machine should be from the same family (all NT, all 2K or all XP). When doing "mixed configuration" certain (OS specific) settings have to be taken into account.

To keep configuration simple it is assumed that two machines have the same operating system installed, are both in the same Domain and have different user accounts logged on. The different users are members of one User Group. This for this User Group access is granted in the DCOM settings of the Client and the Server Machine.



Note: In a mixed configuration (e.g. Server Machine in a Domain and Client Machine in a Workgroup) the so called *double identification* should be done. This means to add User Accounts to the Workgroup Machine which are identically (Name and PWD) the same as the User Accounts on the Domain. By this function calls to the Workgroup Machine can (locally) be identified and granted without "asking" the Domain Controller.

TwinCAT OPC server: DCOM details

Operating System

Depending on the operating system different identification of Users will be performed. Considerations regarding this issues are described in the following pages.

- [Windows XP](#)
- [Windows 2000](#)

TwinCAT OPC server: DCOM details

Windows XP

This section describes the required settings for Windows XP operating system only.

The default installation for XP forces remote users to authenticate as Guest. This means that DCOM clients cannot connect to a server running on an XP machine unless the Guest account is enabled and has enough rights to launch the server. On the other hand when a DCOM server fires a callback to a remote Client installed on an XP machine, the authentication will be "changed back" to the Guest account (which is mostly disabled by default). Thus, the callback (e.g. OnDataChange) will never get though to the Client.

To force an XP machine to "behave" like a Windows 2000 computer the *Network Access* should be changed to **Classic**

Open the Security Options dialog with: *START >> Control Panel >> Administrative Tools >> Local Security Policy >> Local Policies >> Security Options* . Find the following entry: *Network access: Sharing and security model for local accounts* and change this setting to: **Classic - users authenticate as themselves**

Note: in a mixed configuration (e.g. Client installed on XP and Server installed on Windows 2k) the XP machine automatically "changes" to the classic Win2k behavior when launching and accessing the DCOM Server, but there will be no callbacks coming through.

TwinCAT OPC server: DCOM details

Windows 2000

This section describes the required settings for Windows 2000 operating system only.

On Windows 2000 SP 1 operating system there are some known bugs related to proper callback operation. DCOM servers installed on a Win2k SP1 machine sporadically stop sending callbacks (e.g. OnDataChange) after days or weeks. The error code 0x80010108 (RPC_E_DISCONNECTED) is returned by the DCOM Server when trying to fire a callback. All other calls from the Client to the Server still work fine.

There are three ways to overcome this problem:

- release the callback object and subscribe again (recreate the callback object)
- install COM + Rollup Package 18.1 (Post Service Pack 2)
- install Service Pack 3 (or higher) for Windows 2000

TwinCAT OPC server: DCOM details

Client Machine

This section describes the required settings of the Client side only. The Client side is the machine (computer) on which the OPC Client application is running. Normally on this machine no OPC Server is installed. However, on some systems OPC Clients and Servers are installed on the same node, thus the settings of the Server side installation must be performed also.

Depending on the operating system running on the Client machine the appearance of the dialog boxes is different. Generally the same settings must be performed. However, some "operating system specific" settings must be done to get the DCOM security running.

- Windows NT 4 SP 6a
- [Windows 2000 SP 3](#)

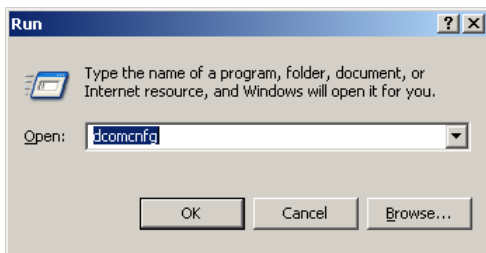
- [Windows XP SP 1](#)
- Windows Server 2003 SP 1

TwinCAT OPC server: DCOM details

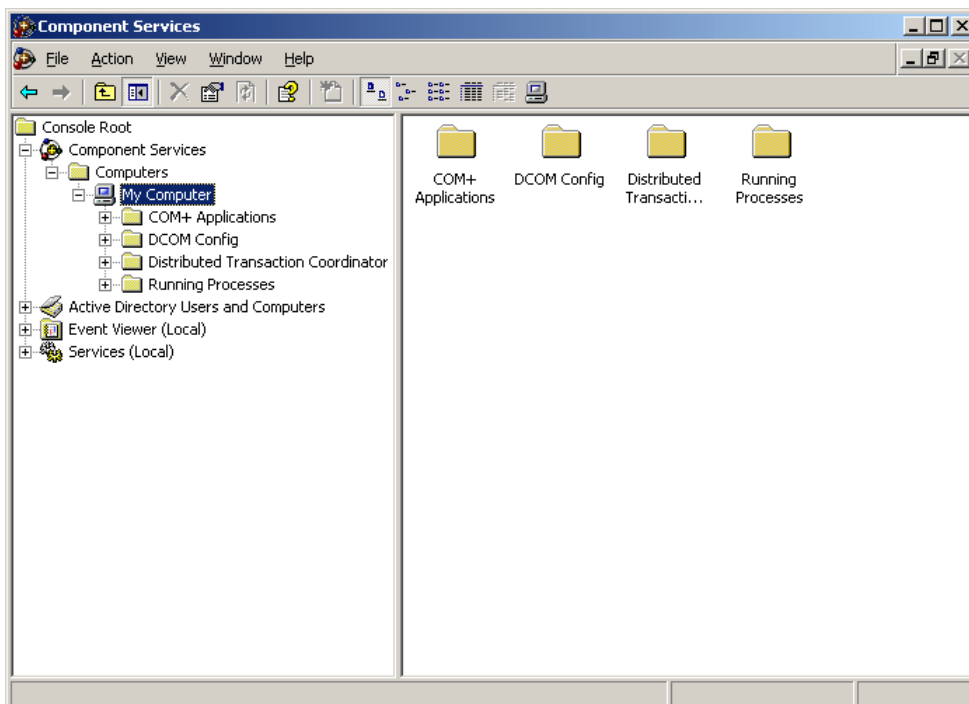
Open DCOM Configuration

Only local administrators are allowed to open and change the DCOM security.

Use *START >> RUN >> dcomcnfg.exe* to start the DCOM configuration dialog.



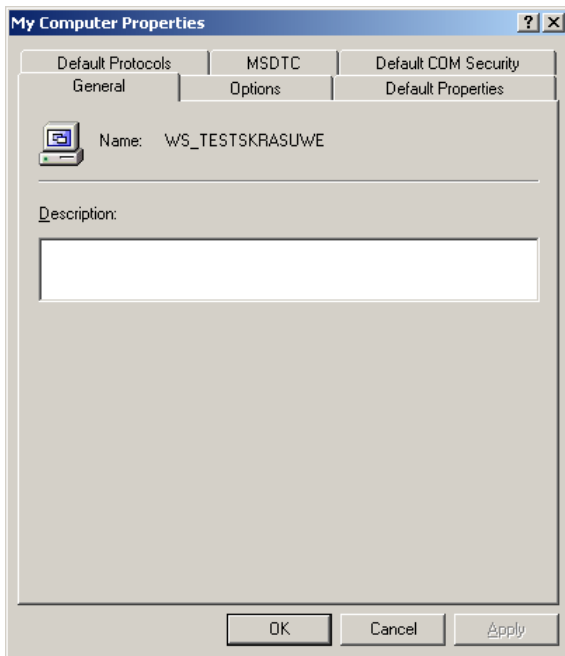
Browse down to *Console Root >> Component Services >> Computers* and right click on *My Computer* than select **Properties** to open the DCOM configuration dialog.



TwinCAT OPC server: DCOM details

General Tab

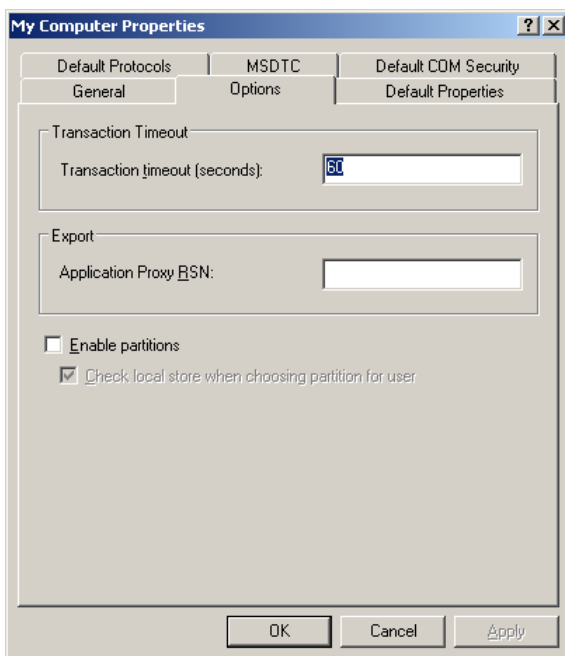
On the General Tab no changes have to be done. The default settings will be correct for OPC Client side security settings.



TwinCAT OPC server: DCOM details

Options Tab

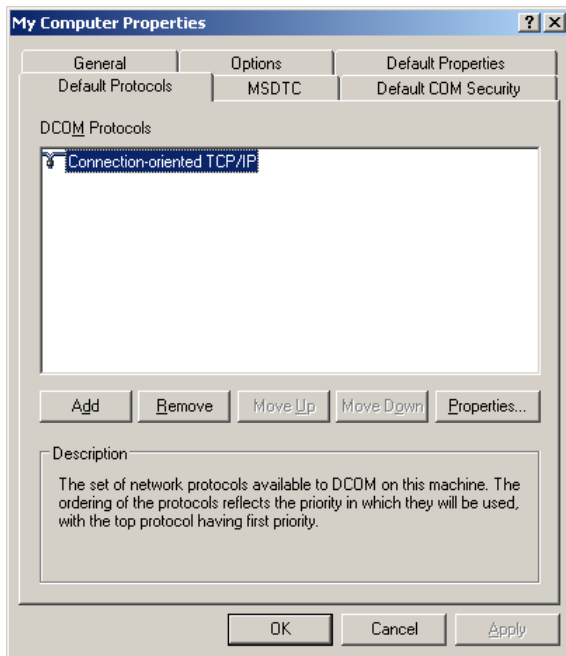
On the Options Tab no changes have to be done. The default settings will be correct for OPC Client side security settings.



TwinCAT OPC server: DCOM details

Default Protocols Tab

On the Default Protocols Tab the *Connection-oriented TCP/IP* protocol should be moved to the top position. This setting forces the use of TCP/IP for DCOM connections. All other protocols can be removed if they are not used with DCOM. The timeout will be reduced if DCOM tries to connect only on TCP/IP connections.



TwinCAT OPC server: DCOM details

Default Properties Tab

On the Default Properties Tab the *Enable Distributed COM on this computer* must be checked. The Authentication Level and the Impersonation Level are set to *Connect* and *Identify* by default.

If the client machine runs in a **Workgroup** the level should be changed

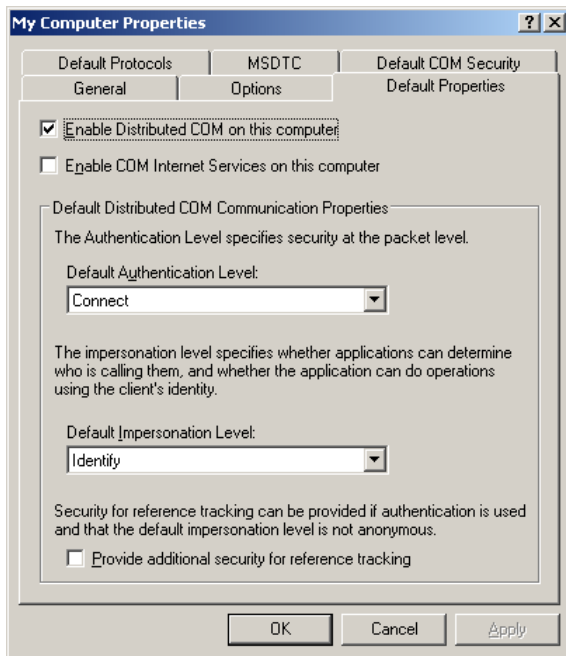
- Authentication Level = None
- Impersonation Level = Anonymous

If the client machine runs in a **Domain** the level remains to default settings

- Authentication Level = Connect
- Impersonation Level = Identify

If the client machine runs in a **mixed configuration** (e.g. the Client machine in a Workgroup and the Server machine in a Domain) the level should be changed. The machine being part of the Domain must be able to identify the security context without "asking" the Domain. Therefore the machine must "know" the users (they must have a local Login).

- Authentication Level = None
- Impersonation Level = Anonymous



Note: not all possible combinations of setting these two levels make sense.

Known Bugs: on Windows 2000 operating systems the Network Configuration Icon disappears when setting DCOM security levels to *None* and *Anonymous*. The network still works but the IP-Address of the NIC can not be changed anymore. Change temporarily to default settings to change IP Address or use *None* and *Delegate*.

TwinCAT OPC server: DCOM details

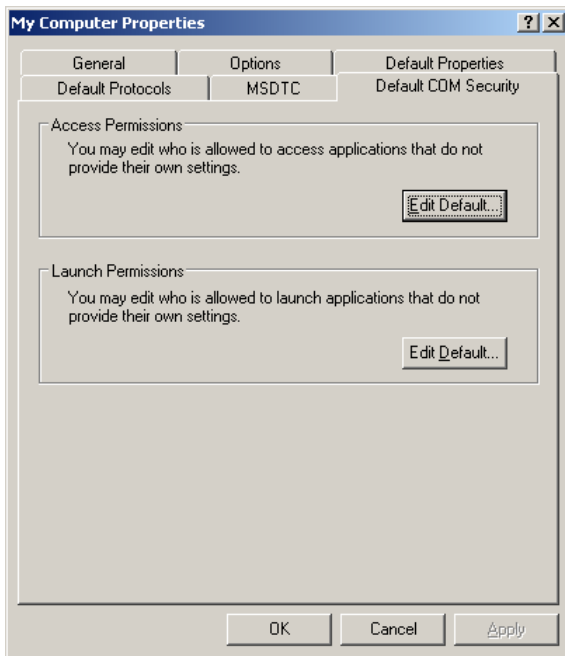
Default COM Security Tab

On the Default COM Security Tab the Access- and Launch permission for all COM-Objects can be changed. As the OPC Client is nothing else than a COM Client, the security settings should be changed to grant access to the Client application. Specially when the OPC Server sends callbacks (e.g. OnDataChange) to the OPC Client the server's process must have access permission on the Client.

The **Default Access Permission** should be granted for

- Administrators
- Interactive User
- System
- Network
- "OPC Server's Security Context"

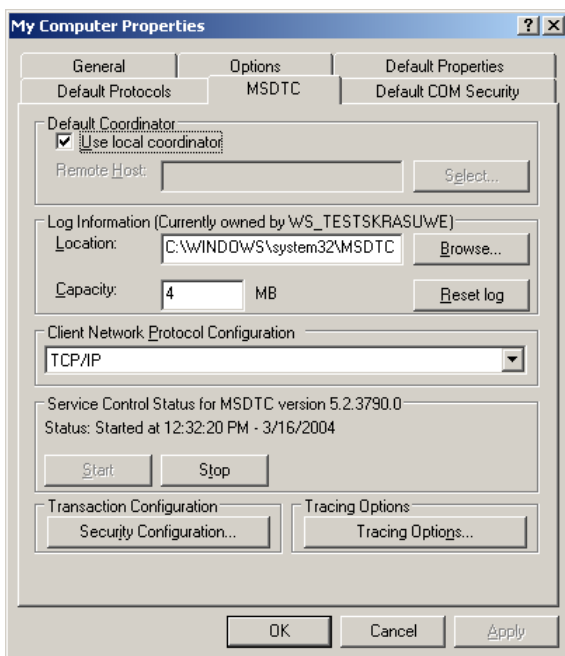
The **Default Launch Permission** should not be changed on the Client machine.



TwinCAT OPC server: DCOM details

MSDTC Tab

On the MSDTC Tab no changes have to be done. The default settings will be correct for OPC Client side security settings.



TwinCAT OPC server: DCOM details

Server Machine

This section describes the required settings of the Server side only. The Server side is the machine (computer) on which the OPC Server application is running. Normally on this machine no OPC Client is installed. However, on some systems OPC Clients and Servers are installed on the same node, thus the settings of the Client side installation must be performed also.

Depending on the operating system running on the Server machine the appearance of the dialog boxes is different. Generally the same settings must be performed. However, some "operating system specific" settings must be done to get the DCOM security running.

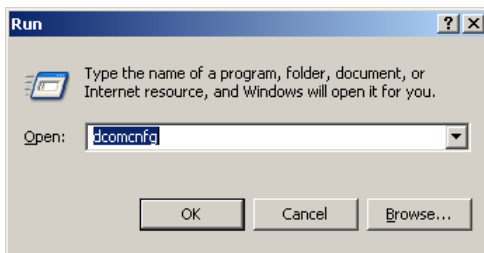
- o Windows NT 4 SP 6a
- o [Windows 2000 SP 3](#)
- o [Windows XP SP 1](#)
- o Windows Server 2003 SP 1

TwinCAT OPC server: DCOM details

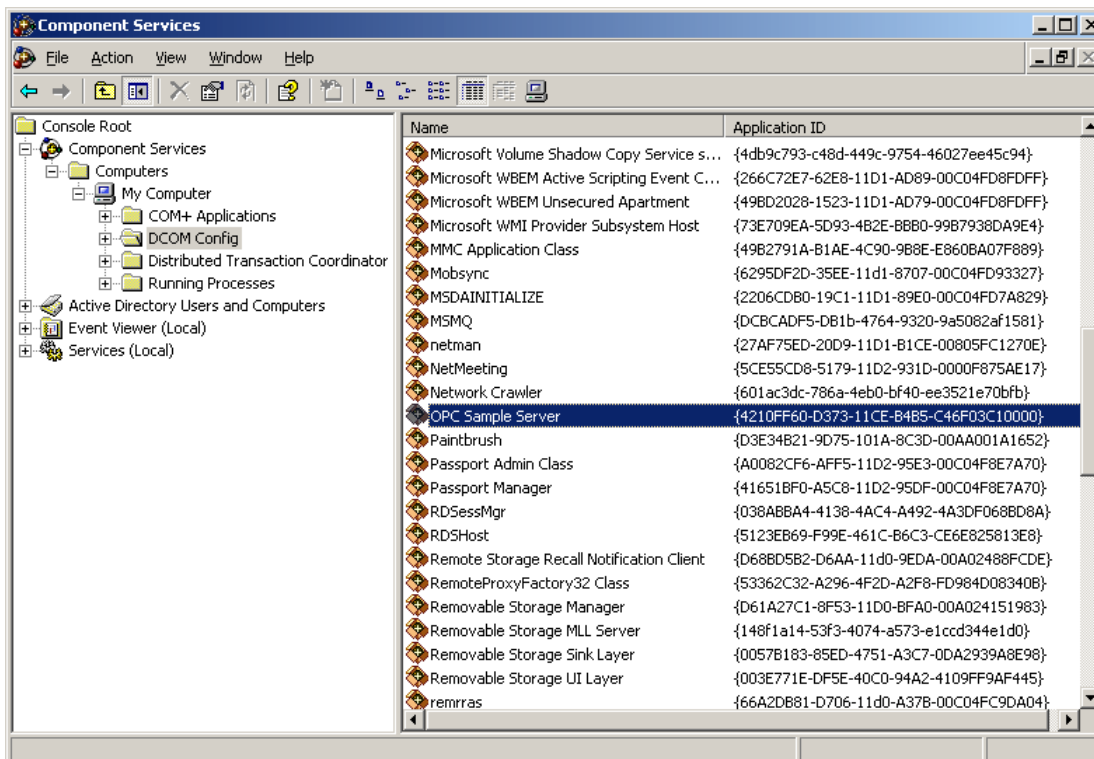
Open DCOM Configuration

Only local administrators are allowed to open and change the DCOM security.

Use **START >> RUN >> dcomcnfg.exe** to start the DCOM configuration dialog.



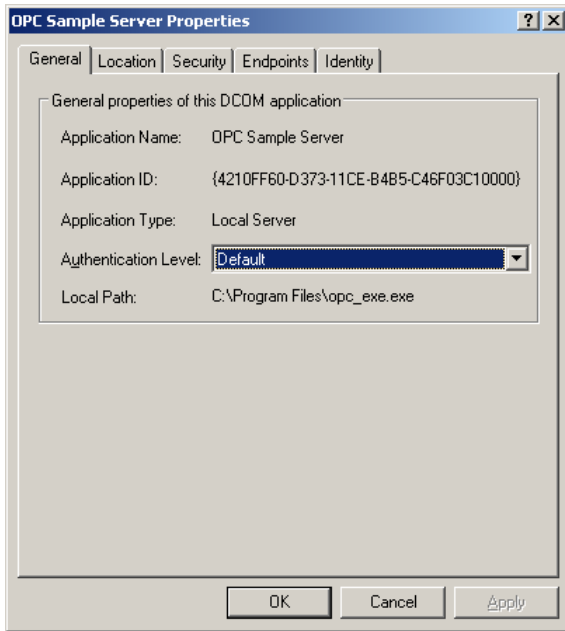
Browse down to *Console Root >> Component Services >> Computers >> My Computer >> DCOM Config* to display all DCOM server applications. Select the appropriate Server and right click on it. Select the **Properties** dialog to change the DCOM security for this specific DCOM Server only.



TwinCAT OPC server: DCOM details

General Tab (server specific)

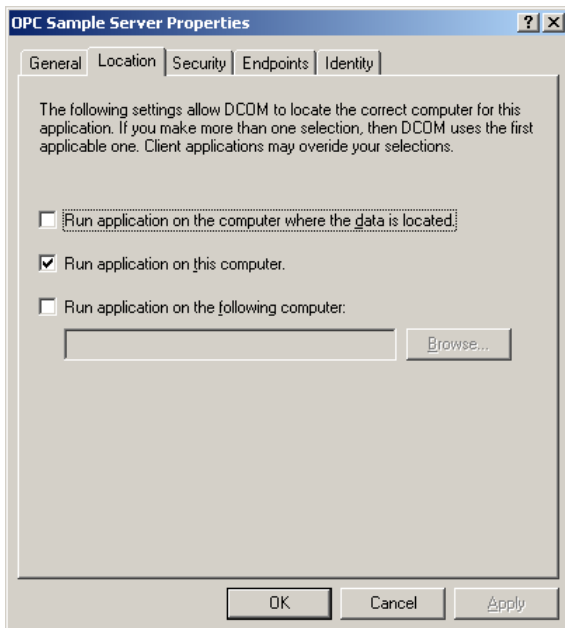
On the server specific General Tab no changes have to be done. The default settings will be correct for OPC Server side security settings. The Authentication Level set to *Default* will overtake the settings from the [Default Properties Tab](#) valid for all COM Objects on this machine (*Connect* by default).



TwinCAT OPC server: DCOM details

Location Tab (server specific)

On the server specific Location Tab no changes have to be done. The default settings will be correct for OPC Server side security settings.



TwinCAT OPC server: DCOM details

Security Tab (server specific)

On the server specific Security Tab the Access- and Launch permission for this specific COM-Server can be changed. As the OPC Server is nothing else than a COM Server, the security settings should be changed to grant access and launch permission to the Server application.

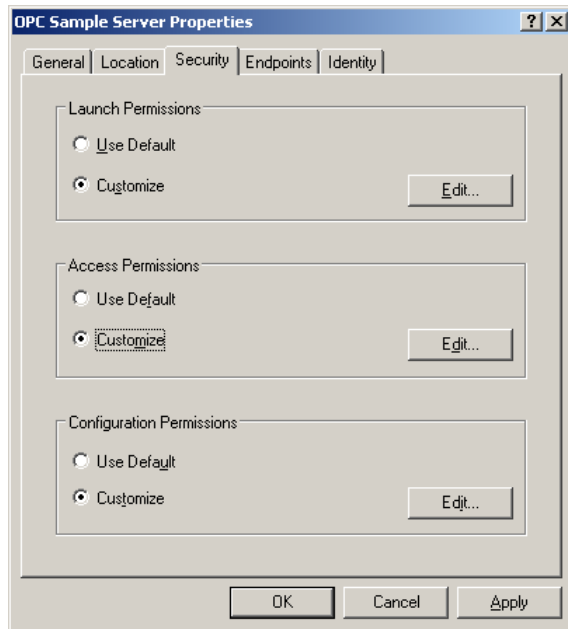
The **Server Specific Access Permission** should be granted for

- Administrators
- Interactive User
- System
- Network
- "OPC Client's Security Context"

The **Server Specific Launch Permission** should be granted for

- Administrators
- Interactive User
- System
- Network
- "OPC Client's Security Context"

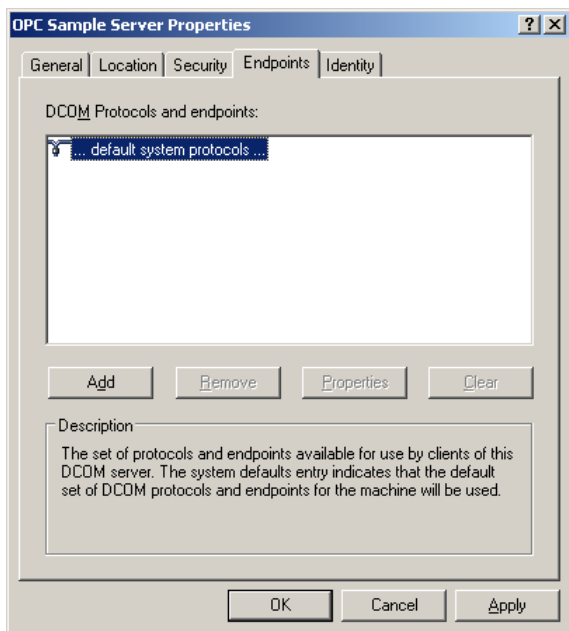
The **Server Specific Configuration Permission** should not be changed on the Server machine.



TwinCAT OPC server: DCOM details

Endpoints Tab (server specific)

On the server specific Endpoints Tab the default setting should remain. In the [Default Protocols Tab](#) for all COM-Objects *Connection-oriented TCP/IP* protocol should be moved to the top position.



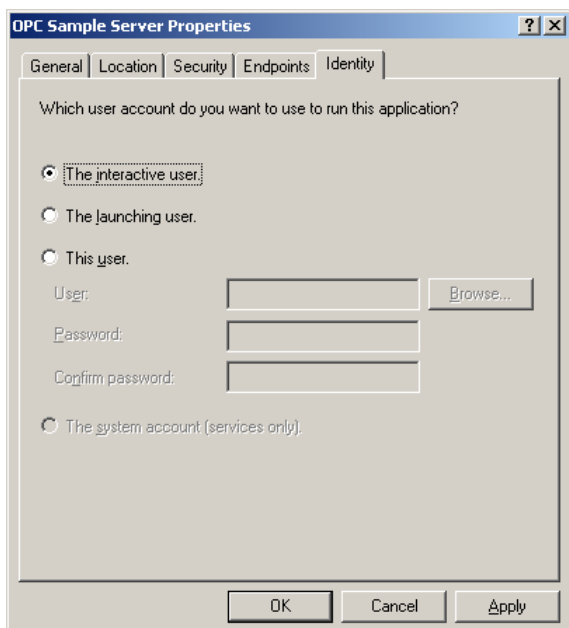
TwinCAT OPC server: DCOM details

Identity Tab (server specific)

On the server specific Identity Tab no changes have to be done. By default the *Interactive User* will be selected. This means that the server will be Launched with the security context of the interactive user (the user that is actually logged on). As this security context changes if somebody else is logged on, in most cases a specific person should be selected i.e even if nobody is logged on to the machine (after reboot) the server can be launched having always the same security context.

The preferred setting should be the third selection.

- The interactive user = default
The interactive user depends on the person that is logged on, thus it can be different each time and only exists if somebody is logged on.
- The launching user = should NEVER be used
The launching user will have the security context of the Client application (the OPC Client launches/connects the OPC Server). When having different Clients in the network, several instances of the Server will be launched having different security context each.
- **This user** = Server will overtake the security context of this person
By selecting *This user* it will be guaranteed that always the same person's security context is used when the server is started. On the Client side only for this person the *Access Permission* must be granted.



TwinCAT OPC server: DCOM details

Permission

Within DCOM Security the launch permission is described as the right to start (launch) an application. As an OPC Server will be started (launched) by the Client application (CreateInstance) the launch permission for the OPC Server application must be granted for the *Security Context* in which the OPC Client is running. The *Security Context* of the Client may be different depending on the user actually logged in and double clicked the Client.

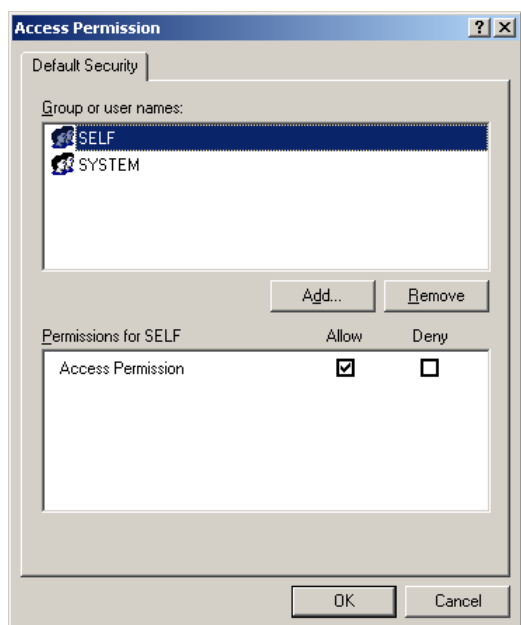
The access permission is described as the right to access (communicate to) an application. As an OPC Client will call functions on the OPC Server Object the *Security Context* of the Client must be known by the Server. When sending callbacks to the Client (asynchronous functions or DataChange) the Client must grant access permission to the OPC Server.

- [Access Permission](#)
- [Launch Permission](#)
- [Select Users](#)

TwinCAT OPC server: DCOM details

Access Permission

For granting *access* permission (accessing the application) to a certain user the *Edit...* button must be clicked opening the following dialog.



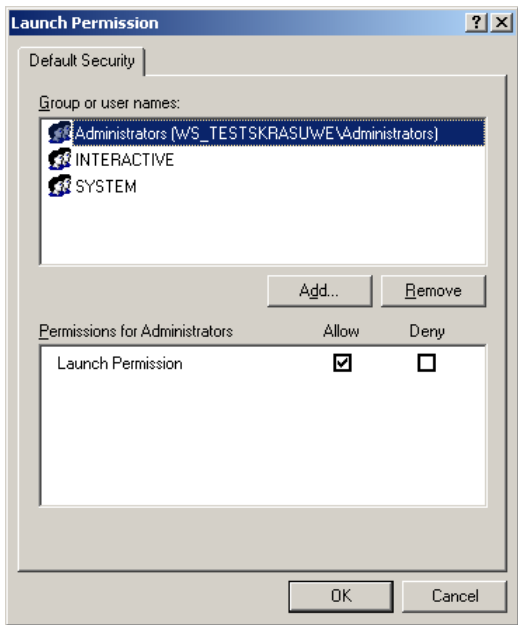
For selecting users click the *Add..* button:

[Selecting Users](#)

TwinCAT OPC server: DCOM details

Launch Permission

For granting *launch* permission (starting the application) to a certain user the *Edit...* button must be clicked opening the following dialog.



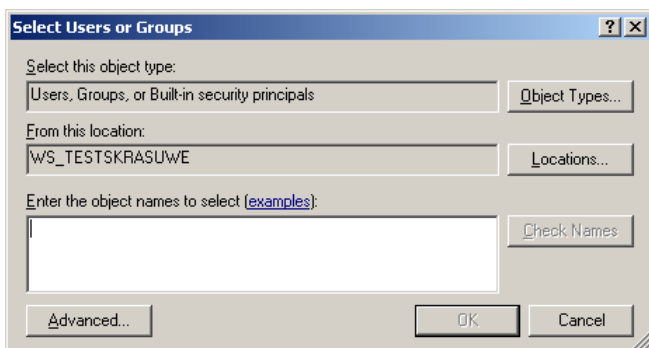
For selecting users click the *Add..* button:

[Selecting Users](#)

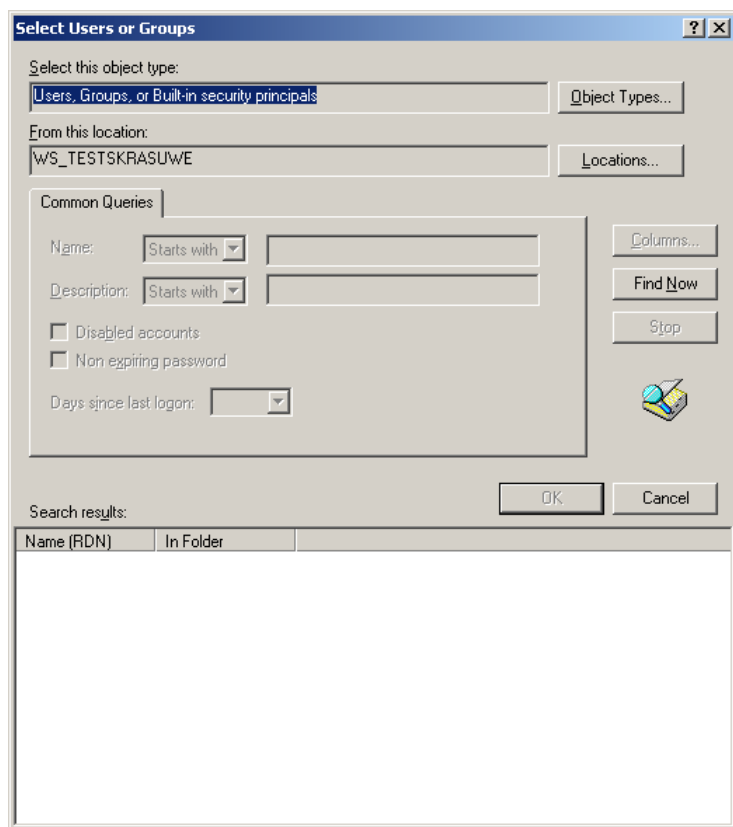
TwinCAT OPC server: DCOM details

Select User

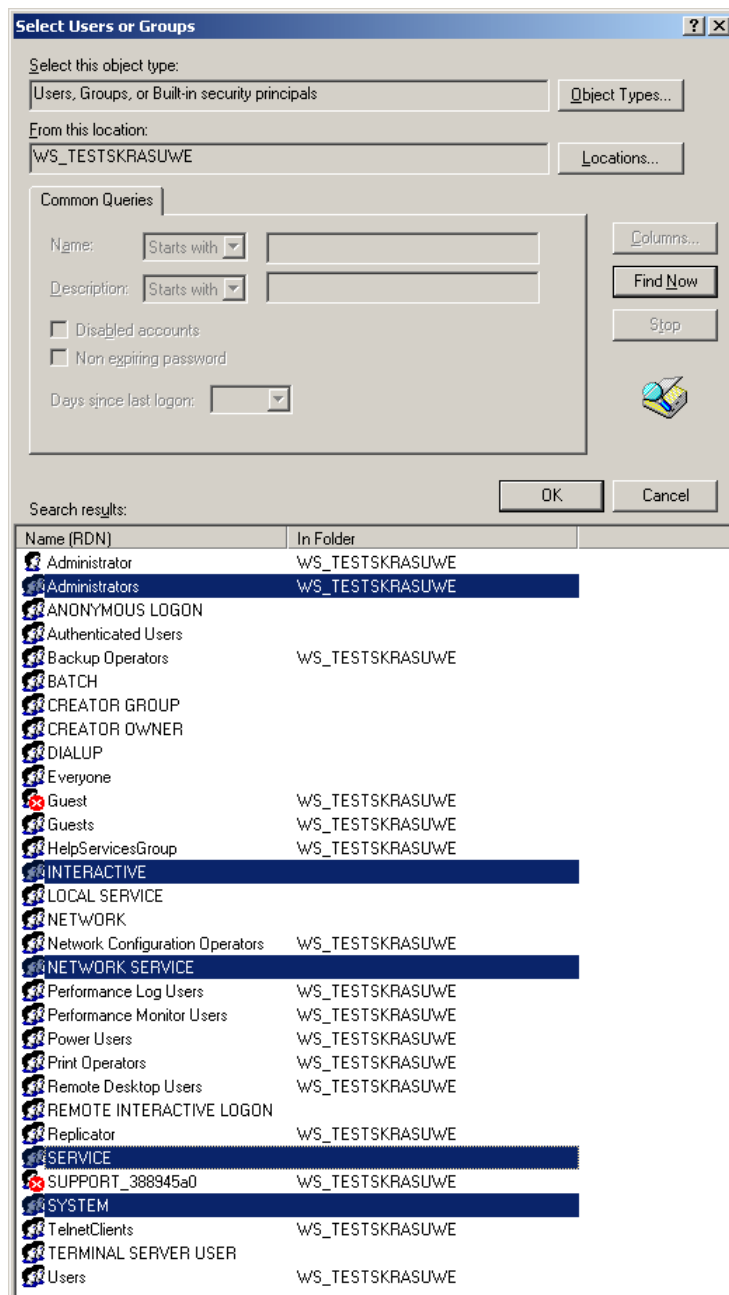
A filter on object types can be set to reduce the listing of all user names. The location states the machine name where the user account is stored, this can be the local workstation or the central domain controller machine. The edit box below the names can be typed in and verified afterwards by pressing the *Check Names* button.



All users known by the machine selected in *Location* can be displayed by pressing the *Advanced...* button.



The list of all known users will be displayed after clicking the *Find Now* button.



TwinCAT OPC XML Server PC

Overview

Prerequisites

TwinCAT OPC-XML-DA requires a PC operating system (Windows 2000 / XP / XPE)
The Microsoft IIS (Internet Information Server) web server is required on that PC.

TwinCAT OPC XML Server PC

Install TwinCAT OPC XML DA on a PC (here: Windows XP)

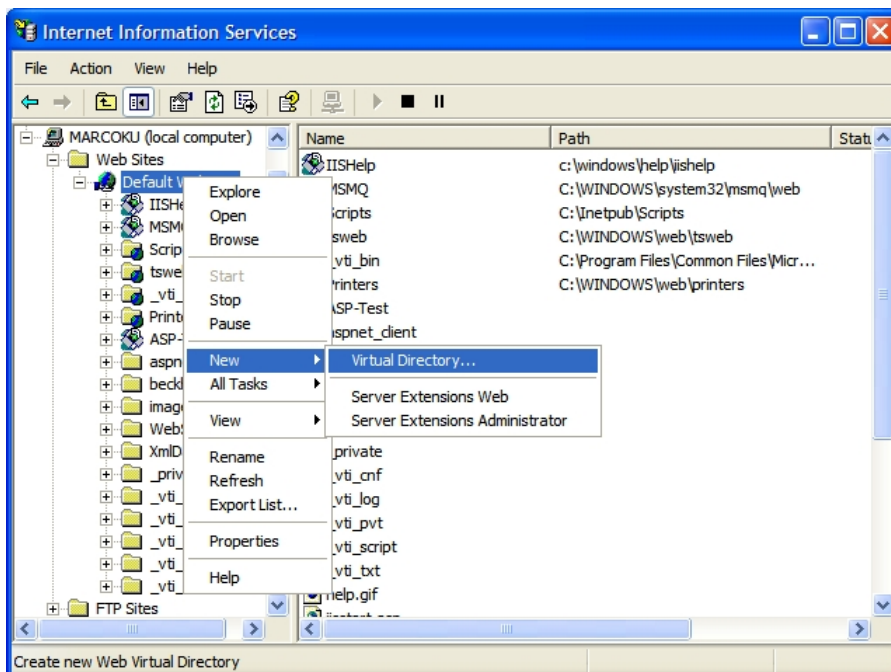
This chapter describes the IIS configuration required for OPC XML DA for operating system XP. (Note : For other Windows-OS the configuration can differ in details).

Requirements :

The TwinCAT OPC XML DA server has to be installed on the PC. This chapter describes how to configure the Microsoft IIS.

Step 1: Create "Virtual Directory" in IIS (Internet Information Service)

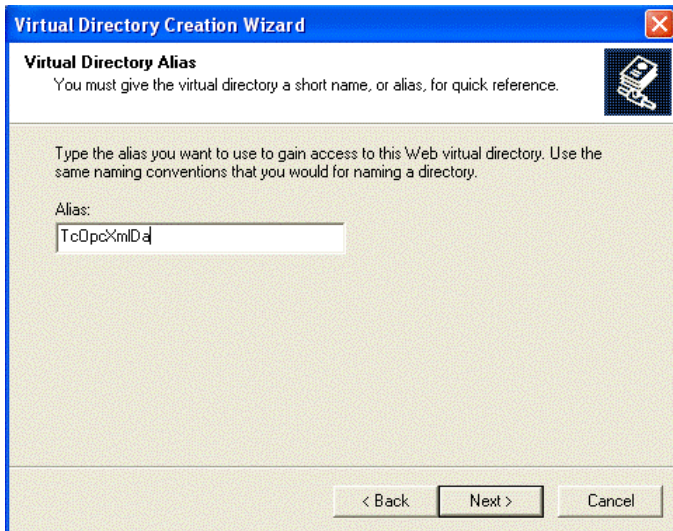
- Open "Internet Information Services" (under "Control Panel/Administrative Tools/").
- Right click on "Default Web Site"
- Select "New" and "Virtual Directory..."



Everybody is welcome, so just click next.

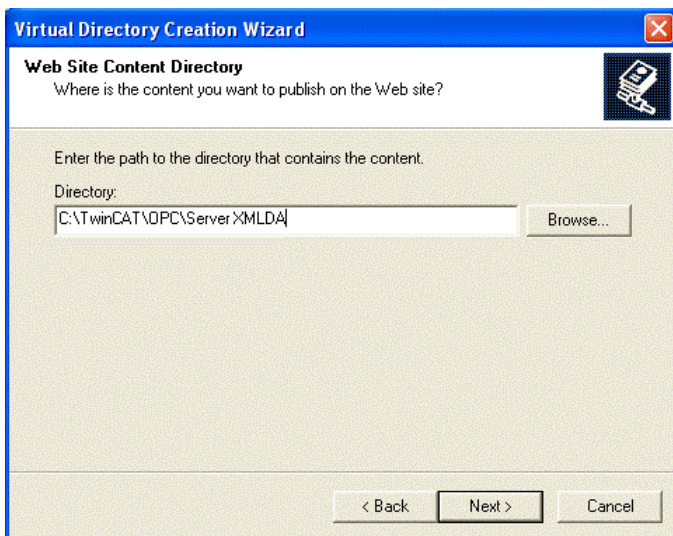


Please enter the alias "TcOpcXmlDa" and click "Next"

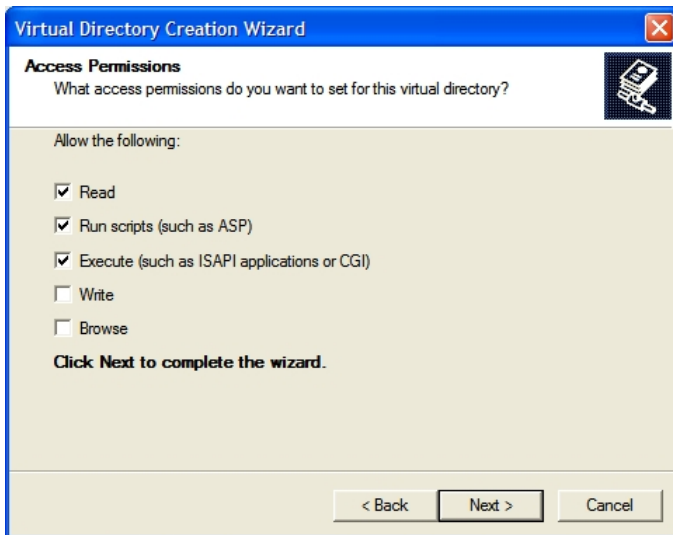


With "Browse..." you have to specify the folder which contains the TwinCAT OPC Xml DA Server. By default the folder should be like "C:\TwinCAT\OPC\Server XMLDA".

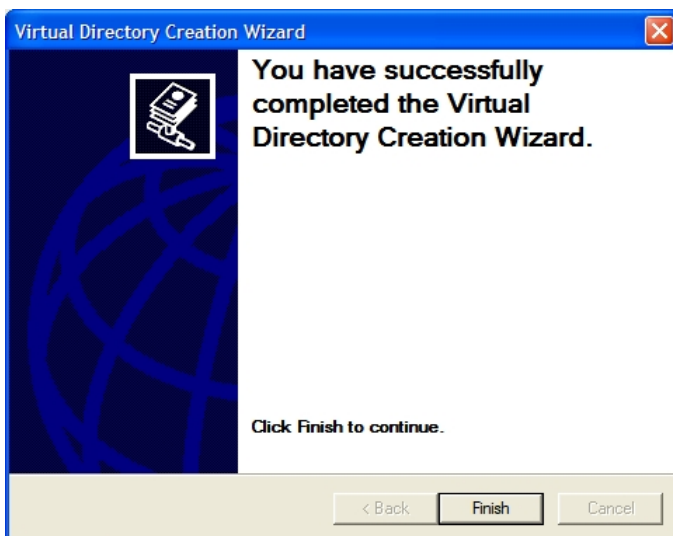
Select "Next" to proceed.



Check options "Read", "Run scripts" and "Execute" and click "Next".



Select "Finish" to finish the configuration of TwinCAT OPC XML DA Server.



Step 2 : Test

The URL of the OPC-XML-DA server on the PC system can be checked locally or from a remote PC: In both cases open the Internet explorer and enter the URL of the OPC Server XML DA on PC system like :

http://<ip-adress or name of PC device>/tcopxmlda/tcopxmlda.dll

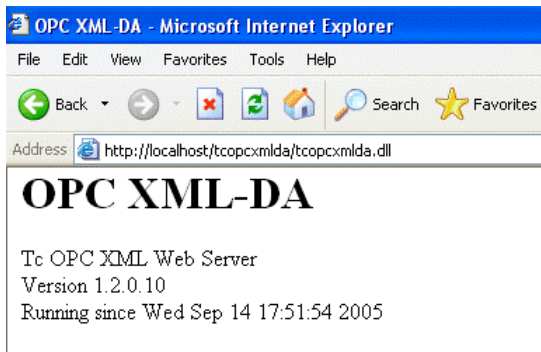
Sample :

http://192.16.17.5/tcopxmlda/tcopxmlda.dll

or

http://localhost/tcopxmlda/tcopxmlda.dll

The TcOpcXmlDa server will reply with a status page containing the product version :

**HINT :**

In case of problems (like receiving no HTML status data) please check if a proxy server is activated on your host PC. After deactivating the proxy and reloading the URL the Opc XML DA server should reply with status info.

TwinCAT OPC XML DA PC server: Configuration

Configuration

This chapter describes how to configure the OPC XML DA Server on BECKHOFF PC based systems.

Summary :

The OPC-XML DA server is configured with default xml-configuration file. Just activate once a checkbox in PLC-Control to download the symbol information to target device.

With TwinCAT 2.10 Build 1240 (or higher) there is no manual configuration required. Please refer to section **"Zero configuration"** on this page.

With previous TwinCAT versions please refer to section **"Manual or specific configurations"** on this page.

Zero configuration (requires TwinCAT 2.10 on host PC)

Details :

- The OPC-XMLDA configuration file "**...\wwwroot\TcOpcXmlDa\TcOpcXmlSvrCfg.xml**" contains the list of ADS-devices which are available via OPC-XML-DA interface.
- By default this list is pre-configured for one local PLC-runtime system (first PLC runtime-system on ADS-port 801)
- By default the required symbol-information is linked to **C:\TwinCAT\Boot\CurrentPlc_1.tpy**

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<TcOpcXmlSvrConfiguration>
<Namespace>
  <Devices>

    <Device>
      <Name>PLC1</Name>
      <AdsNetId>0.0.0.0.0.0</AdsNetId>
      <AdsPort>801</AdsPort>
      <AdsTimeout>2000</AdsTimeout>
      <AdsTimeSuspend>20000</AdsTimeSuspend>
      <AutoCfg>7</AutoCfg>
      <!-- NOTE: CurrentPlc_1.tpy will be downloaded from PLC-Control -->
```

```

    <AutoCfgSymFile>C:\TwinCAT\Boot\CurrentPlc_1.tpy</AutoCfgSymFile>
    <Disabled>0</Disabled>
  </Device>

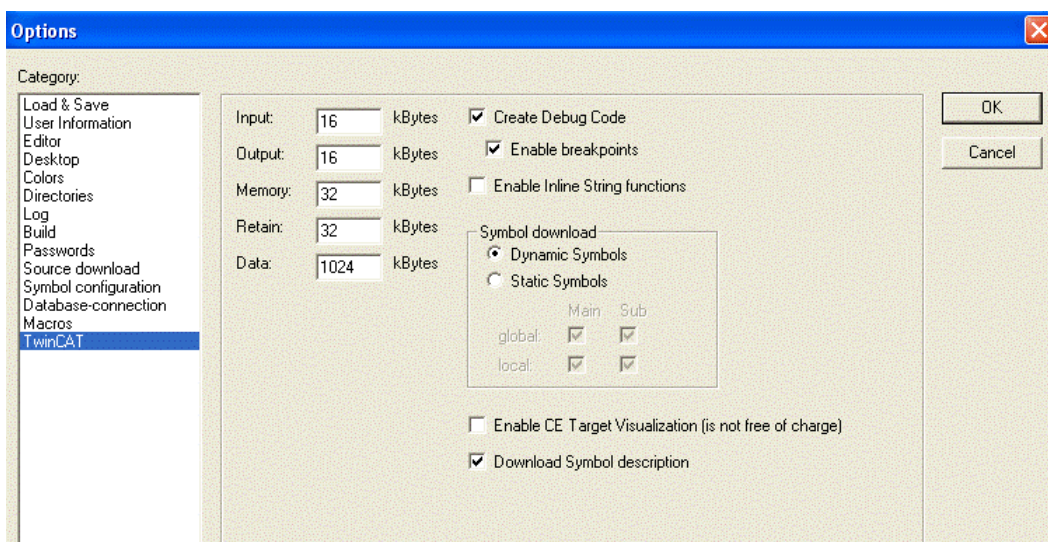
</Devices>
</Namespace>
</TcOpcXmlSvrConfiguration>

```

With TwinCAT Version 2.10 Build 1240 the PLC Control allows to download the symbol file <plc-project-name>.TPY to the target TwinCAT machine. The TPY file contains information about all available PLC-variables in the PLC-project. OPC-XML DA CE Server requires this TPY-information.

To download the current symbol information in a TPY file navigate in the PLC-Control to "Project-->Options-->TwinCAT" and check the option **"Download Symbol description"**

With each download of the PLC-code now also the <project-name>.TPY file is downloaded as "C:\TwinCAT\Boot\CurrentPlc_1.tpy" containing information about symbols of first run time system.
(Second runtime system would be "C:\TwinCAT\Boot\CurrentPlc_2.tpy" etc.)



Summary:

The OPC-XML DA server is configured with default xml-configuration file. Just activate the checkbox to download the symbol information within PLC-Control environment.

NOTE:

OPC-XML DA Server checks configuraton on startup. See notes to [reload configuration](#) today (without reboot)

Optional manual or specific configurations ?

If the default configuration is not available you have to configure the opc-xml-da configuration file "...\\wwwroot\\TcOpcXmlDa\\TcOpcXmlSvrCfg.xml" manual.

- If required the opc-xml configuration file can be done with TwinCAT OPC COnfigurator

Sample for OPC XML DA server accessing multiple ADS devices

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<TcOpcXmlSvrConfiguration>
  <Namespace>
    <Devices>

      <Device>
        <Name>PLC1</Name>
        <AdsNetId>0.0.0.0.0.0</AdsNetId>
        <AdsPort>801</AdsPort>
        <AdsTimeout>2000</AdsTimeout>
        <AdsTimeSuspend>20000</AdsTimeSuspend>
        <AutoCfg>7</AutoCfg>
        <!-- NOTE: CurrentPlc_1.tpy will be downloaded from PLC-Control -->
        <AutoCfgSymFile>C:\TwinCAT\Boot\CurrentPlc_1.tpy</AutoCfgSymFile>
        <Disabled>0</Disabled>
      </Device>

      <Device>
        <Name>PLC2</Name>
        <AdsNetId>0.0.0.0.0.0</AdsNetId>
        <AdsPort>811</AdsPort>
        <AdsTimeout>2000</AdsTimeout>
        <AdsTimeSuspend>20000</AdsTimeSuspend>
        <AutoCfg>7</AutoCfg>
        <!-- NOTE: CurrentPlc_2.tpy will be downloaded from PLC-Control -->
        <AutoCfgSymFile>C:\TwinCAT\Boot\CurrentPlc_2.tpy</AutoCfgSymFile>
        <Disabled>0</Disabled>
      </Device>

    </Devices>
  </Namespace>
</TcOpcXmlSvrConfiguration>
```

NOTE:

OPC-XML DA Server checks configuraton on startup. Reconfiguration might require a new start today.

TwinCAT OPC XML DA CE server

Nice to know

This chapter describes nice helpful hints for OPC-XMLDA server (XP and CE)

OPC specification describes the interfaces which an OPC-XMLDA server have to (MUST) implement.

As an addOn the TwinCAT OPC-XMLDA server provides additional "nice to have" functionalities..

These additional functions are initiated via the URL of the OPC-XML-DA server.

Open the Internet explorer and enter the URL of the OPC Server XML DA on target system like :

http://<ip-adress or name of device>/TcOpcXmlDa/TcOpcXmlDa.dll<Services>

<Services> could be like this :

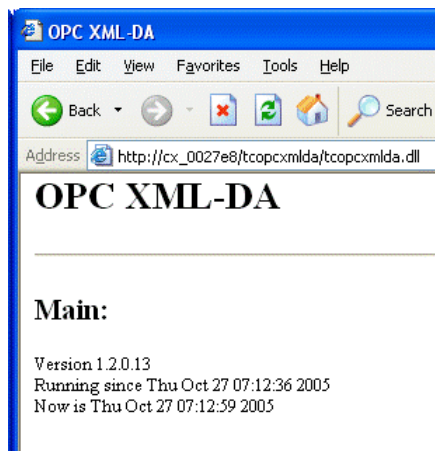
- Status information
- Log information
- Reload configuration
- Display configuration
- Restart server
- Stop server

1. Status information

<Services> = "" (nothing)

SAMPLE : <http://<ip-adress or name of device>/TcOpcXmlDa/TcOpcXmlDa.dll>

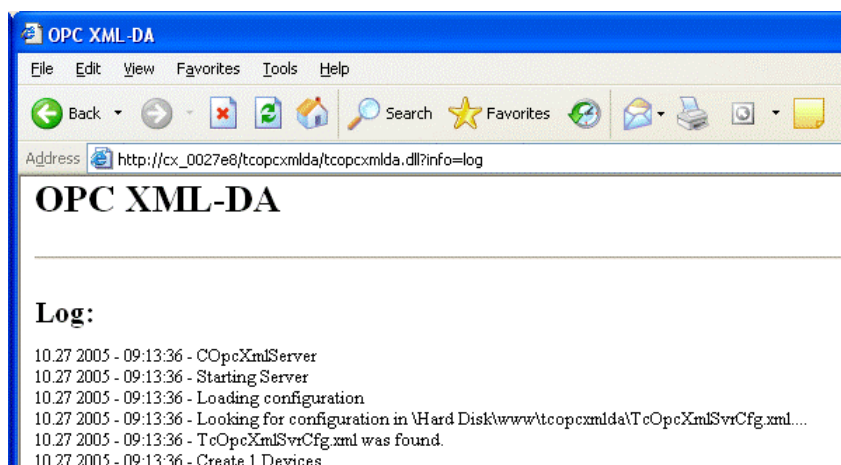
The TcOpcXmlDa server will reply with a status page containing the product version :



2. Log information

<Services> = ?info=log

Sample : <http://<ip-adress or name of device>/TcOpcXmlDa/TcOpcXmlDa.dll?info=log>



3. Reload configuration file

<Services> = ?info=reload

Sample : <http://<ip-adress or name of device>/TcOpcXmlDa/TcOpcXmlDa.dll?info=reload>

4. Display configuration

<Services> = ?info=config

Sample : <http://<ip-adress or name of device>/TcOpcXmlDa/TcOpcXmlDa.dll?info=config>

NOTE : not implemented now, under construction

1. Restart of OPC-XMLDA server

<Services> = ?action=restart

Sample : <http://<ip-adress or name of device>/TcOpcXmlDa/TcOpcXmlDa.dll?action=restart>

2. Stop of OPC-XMLDA server

<Services> = ?action=stop

Sample : <http://<ip-adress or name of device>/TcOpcXmlDa/TcOpcXmlDa.dll?action=stop>

NOTE : not implemented now, under construction

TwinCAT OPC Server CE

TwinCAT OPC Server CE : Optimization

This article describes how to minimize the OPC-namespace of TwinCAT-OPC-Server CE.

Take care to NOT load all PLC variables into OPC-Server namespace but instead load that smaller list of PLC variables which are required via OPC in the OPC-client.

Note :
CE is designed for small systems with limitted resources like CPU and memory. Due to this the optimization of namespace is highly recommended for OPC-Servers running on CE.

NOTE

In general the opc-server is configured by the OPC-XML configuration file to find the detailed symbol information for each ADS device.

- The option <AutoCfg>7</AutoCfg> is designed to import all variables defined in TPY files
 OPC DA : Limitation today : Only global variables (Strings / ints / Bools..) are imported. Today no structures are imported.
 OPC XMLDA : No limitation
- The option <AutoCfg>8</AutoCfg> is designed to import just a subset of PLC variables marked in the PLC-code to be available for OPC..
 OPC DA : Limitation today : Only global variables (Strings / ints / Bools..) are imported. Today no structures are imported.
 OPC XML : No limitation

In this sample the PLC variable "bMemAlarm"1 will not be available via OPC-Server.

The variable "iReadOnly" is marked to be an OPC-variable and is available for OPC-clients. Additional the OPC-Item Property 5 (defined by OPC-Foundation) is set to "1" which means, the opc-server will handle this variable as "Read-Only" . OPC-Server will take care, that OPC-clients can not write to that PLC-variable.

Extra comments in PLC-IEC1131 Editor will modify the configuration of opc-server:

```

bMemFlag1      AT%MX10.0      :   BOOL;  (*~ (OPC : 1 : available for OPC-clients) *)
bMemFlag2      AT%MX10.1      :   BOOL;  (*~ (OPC : 1 : available for OPC-clients) *)
bMemAlarm1     AT%MX10.2      :   BOOL;
bMemAlarm2     AT%MX10.3      :   BOOL;  (*~ (OPC : 1 : available for OPC-clients) *)
iReadOnly      :   INT;        (*~ (OPC : 1 : available fot OPC-clients)
                                (OPC_PROP[0005] : 1 : OPC_PROP_RIGHTS, here ReadOnly) *)

```